



## Introduction to

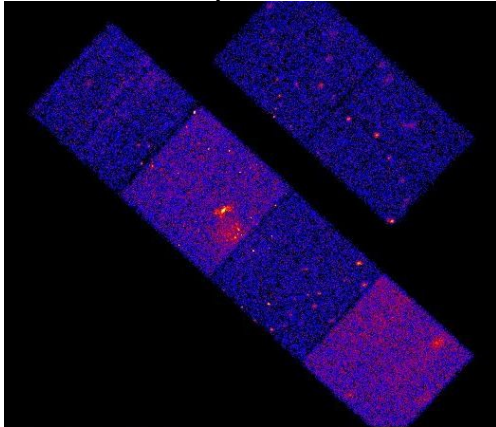


Jonathan McDowell  
Chandra X-ray Center, SAO

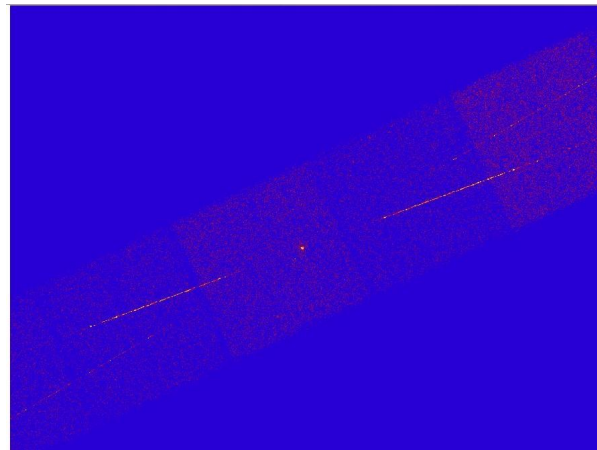
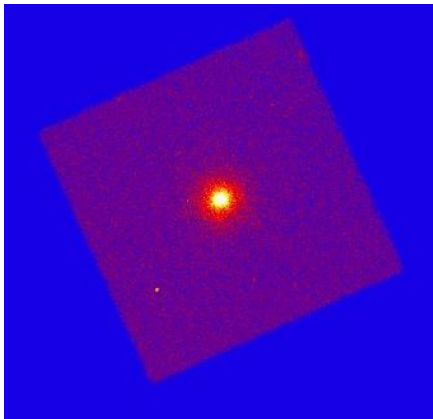
# Scope

- Caveat: will cover ACIS imaging data only

– (81% of recent observations)



- Basics the same for HRC and gratings, but with extra wrinkles





- CXC Home
- Proposer
- Archive
- Data Analysis
- Instruments & Calibration
- For the Public

Google Custom Search Search x  
Search the CIAO website or [contact the CXC HelpDesk](#)

Last modified: 25 August 2014



### CIAO 4.6 Homepage

#### Introduction

- [Welcome](#)
- [Tools & Applications](#)
- [CIAO News](#)

Updated: 4 Nov 2014

#### Data Analysis

- [Analysis Guides](#)
- [Science Threads](#)
- [Why Topics](#)
- [Help Pages \(AHELP\)](#)
- [Video Demos and Tutorials](#)

#### Documentation

- ["Watch Out" List](#)
- [Bug List](#)
- [Frequently Asked Questions \(FAQ\)](#)
- [Manuals & Memos](#)
- [Dictionary](#)
- [Publications](#)
- [Download the Website](#)

#### Download CIAO

- [Download CIAO 4.6](#)
- [Download CALDB](#)
- [Scripts & Modules Package](#)
- [System Requirements](#)
- [Installation Instructions](#)
- [Platform Support](#)
- [Release Notes](#)
- [Version History](#)
- [Other Analysis Software](#)

#### Sherpa (Modeling and Fitting)

- [Sherpa website](#)
- [Threads](#)
- [Help Files](#)

#### ChIPS (Plotting Package)

- [ChIPS website](#)
- [Threads](#)
- [Help Files](#)

#### Scripting in CIAO

- [Introduction](#)
- [Modules](#)
- [Running Tools from Python](#)

#### Data Products



## Chandra Interactive Analysis of Observations

*from "s'sciavo", "I am your servant" in Venetian dialect.*

### WHAT'S NEW | WATCH OUT

Quick CIAO links: [Download CIAO](#) | [Science Threads](#) | [Help Files](#) | [Scripts](#)

Related CXC sites: [ChIPS](#) | [Sherpa](#) | [PSFs with ChaRT](#) | [CALDB](#)

[Chandra Source Catalog](#) | [TGCat: Chandra Gratings Catalog](#)

**Install  
CIAO 4.6.1  
& CALDB 4.6.3**

Read the [CIAO 4.6.1 release notes](#) for detailed information on this release, including [How CALDB 4.6.3 Affects Your Analysis](#).



Subscribe to the CIAO News RSS feed



Subscribe to Chandra/CIAO announcements

#### I've never used CIAO before. Where should I begin?

- [Welcome to CIAO](#)
- [Introduction to the Tools & Applications](#)

- [Download CIAO 4.6.1](#)
- [Installing CIAO 4.6.1 thread](#)

[Introductory Science Threads](#)

[Analysis Guides](#)

#### Has a new version of CIAO or CALDB been released? What has changed in the site recently?

[What's New](#)

["Watch Out" List](#)

[Version History](#)

[CIAO Release Notes](#)

[CALDB Release Notes](#)

# CIAO: supports users from proposal to publication

## Tools for proposal planning

Assessing feasibility and examining Chandra field-of-view



e.g. *obsvis*, *colden*

## Tools for data discovery and access

Command line programs to complement the Archive and Catalog searches



e.g. *find\_chandra\_obsid*,  
*download\_chandra\_obsid*,  
*search\_csc*

## Tools for data inspection and exploration

What is this dataset? How many photons? What instrument configuration? Quick look visualization...



e.g. *ds9*, *prism*, *dmlist*,  
*dmstat*, *dmcoppy*

## Tools for data reduction

Apply latest calibrations to observation  
Locate sources and measure their properties (position, brightness)  
For each source, generate tailored calibration files (e.g. spectral calibration)



e.g. *dmextract*, *wavedetect*,  
*specextract*, *srcflux*,  
*fluximage*

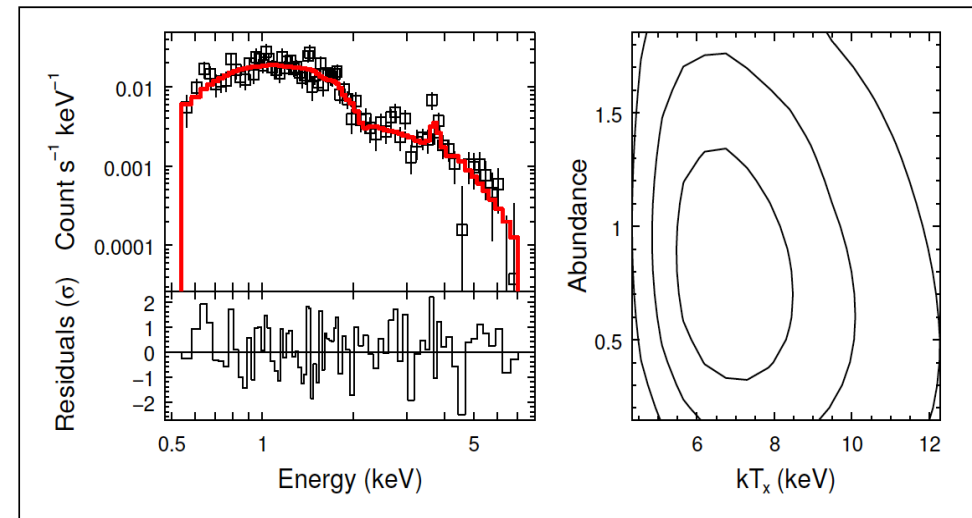
## Tools for interactive data analysis

'sherpa' – 1D and 2D modeling and fitting  
Python environment – familiar to the new generation of astronomers and used in other missions



## Tools to prepare for publication

'ChIPS' – publication quality graphics integrated with the analysis system







# CALDB

- The CALDB (Calibration Database) contains everything you need that's not part of your specific observation.
- It's designed as a multimission directory structure. The Chandra files are in `$CALDB/data/chandra`
- Within that, they are arranged by instrument and kind of calibration. But, with luck, the software will find the CALDB files you need automatically.
- Just make sure that you keep the CALDB up to date! But, be careful - if you start off processing with a given version of the CALDB and CIAO, then upgrade to a new CALDB and CIAO, things are sometimes incompatible. Check the release notes.
- **NEW FEATURE!** *download\_obsid\_caldb* will download just those CALDB files needed for the observation you are working on. Useful if your internet connection is poor.

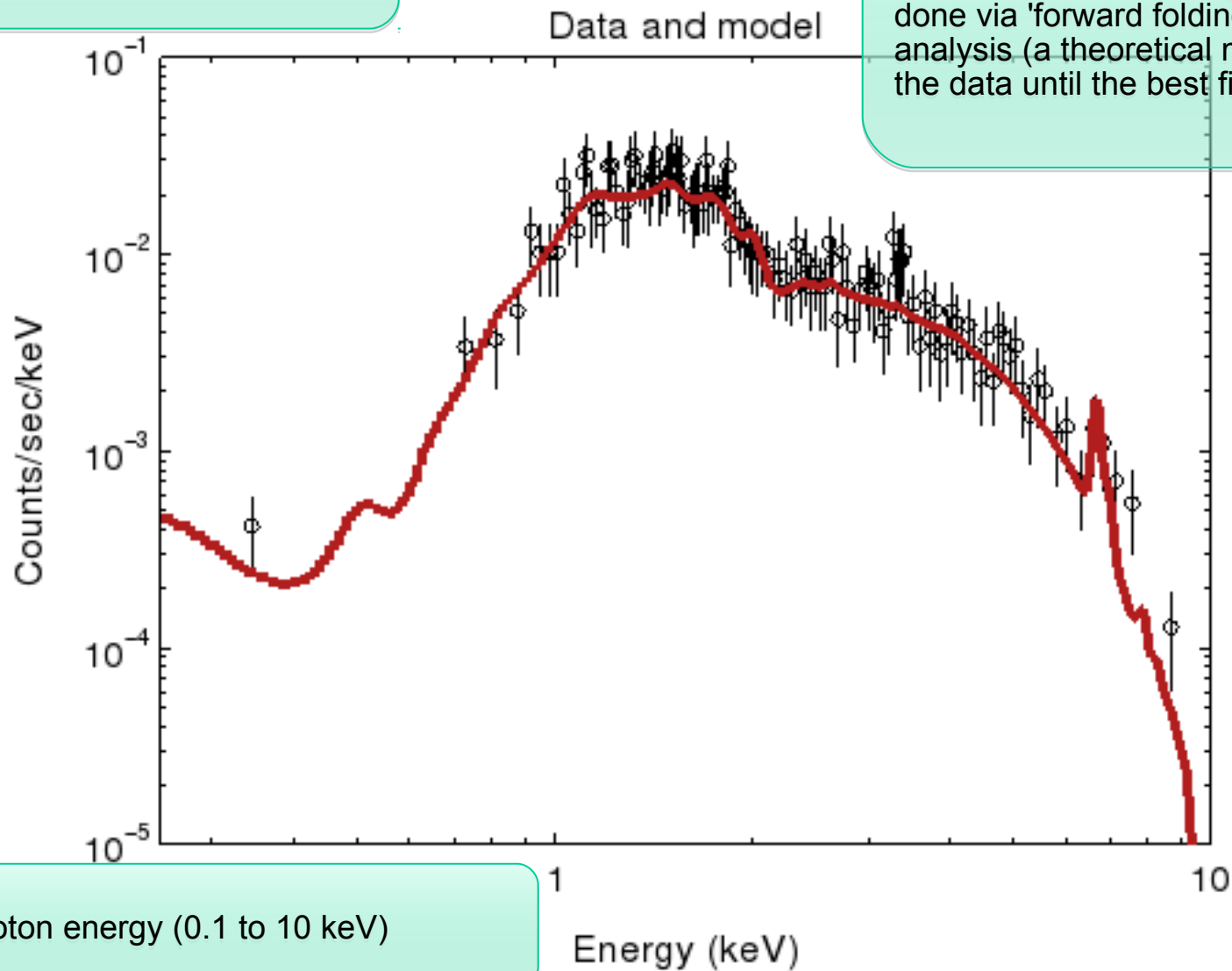
# Introduction to X-ray Data Analysis

- X-ray astronomy is different .....
- **Problem 1:** Photon counting with small number statistics
- **Problem 2:** Spectral line spread function is often broad and messy - forced to forward-folding approach
- **Problem 3:** Bands are very broad, so energy (wavelength) dependence more obvious (e.g. in PSF)
- **Problem 4:** Different optics - PSF degrades rapidly off axis
- **Problem 5:** The telescope is not pointing steadily like, say, HST - it's moving back and forth across the source.
- But:
- **Advantage:** We have more information on each photon (position, energy, arrival time)

# Complexities in X-Ray and Chandra Data Analysis

In Poisson statistics regime because of the small number of photons

Imaging data has limited energy resolution and modeling can only be done via 'forward folding' spectral analysis (a theoretical model is fitted to the data until the best fit is found)



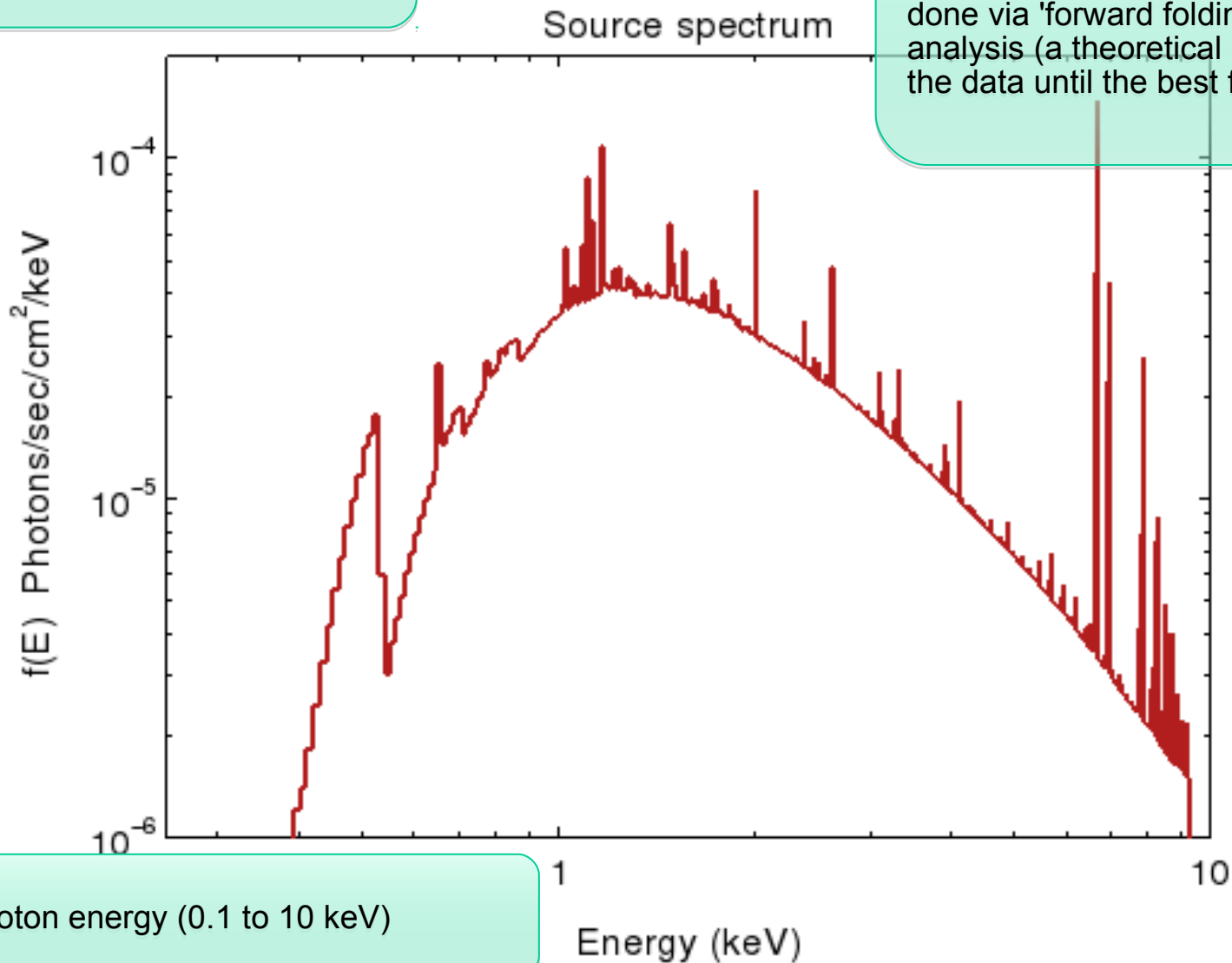
2 decades of photon energy (0.1 to 10 keV)



# Complexities in X-Ray and Chandra Data Analysis

In Poisson statistic regime because of the small number of photons

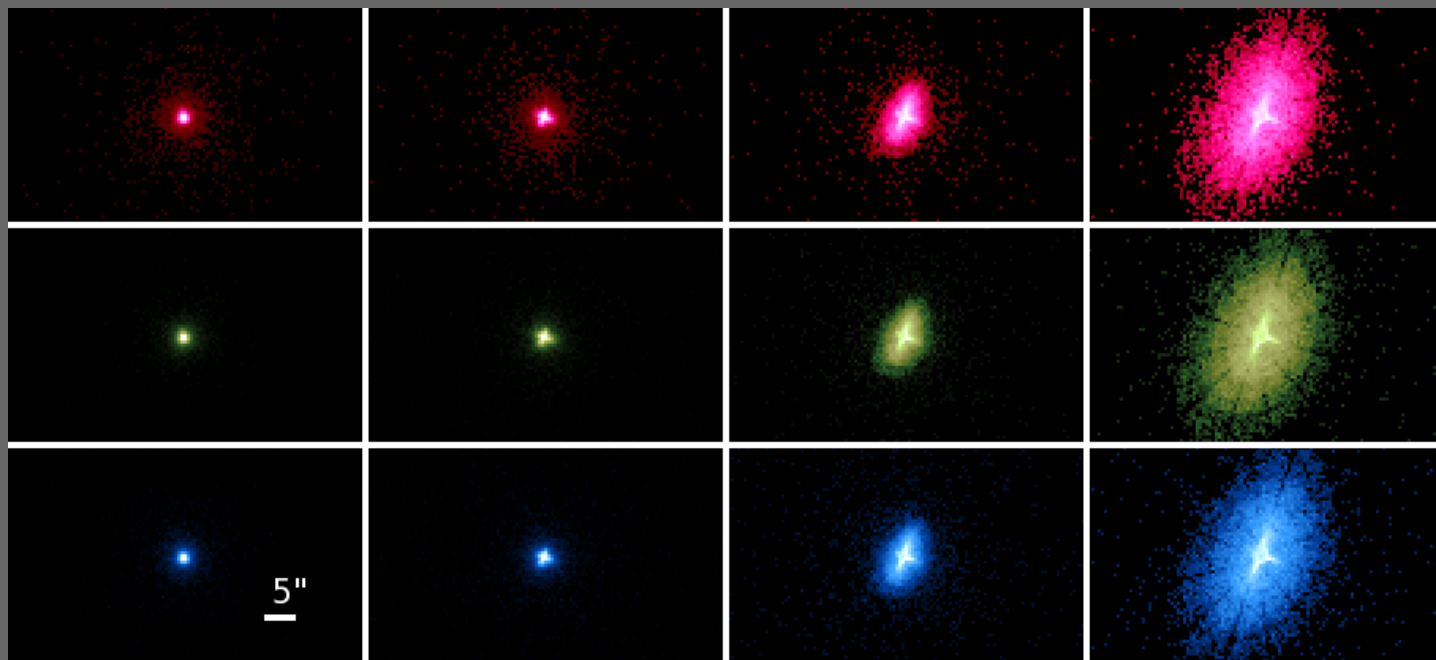
Imaging data has limited energy resolution and modeling can only be done via 'forward folding' spectral analysis (a theoretical model is fitted to the data until the best fit is found)



2 decades of photon energy (0.1 to 10 keV)

# Complexities in X-Ray and Chandra Data Analysis

The Chandra PSF



Every aspect of the observation varies with

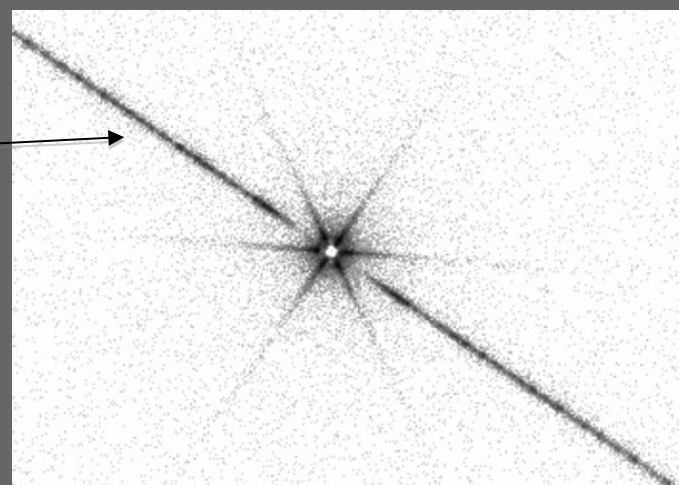
- energy
- position
- time

(e.g. image sharpness, sensitivity, instrumental energy scale)



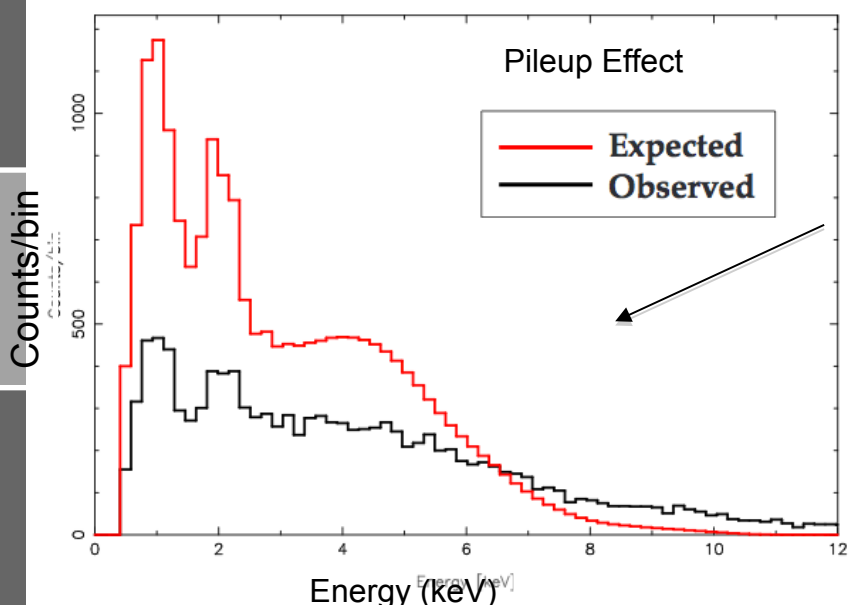
Energy

Off-axis angle



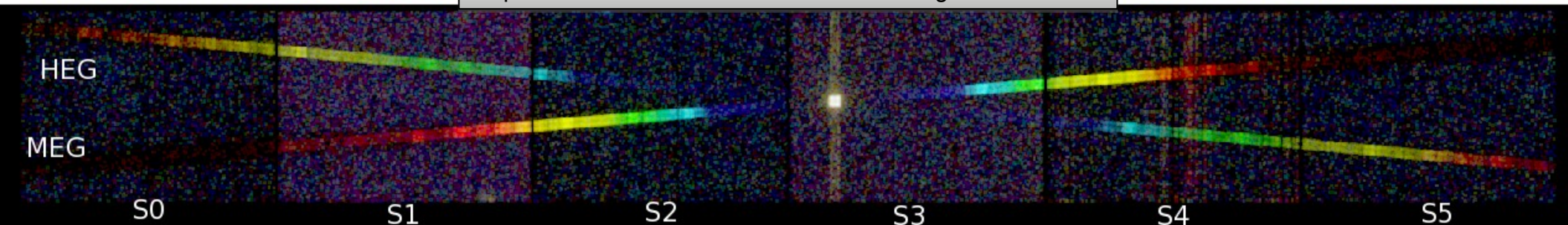
Specific instrumental effects eg.  
 readout streak  
 pileup - two or more photons detected as single event

Q0836+7104 predicted vs. observed

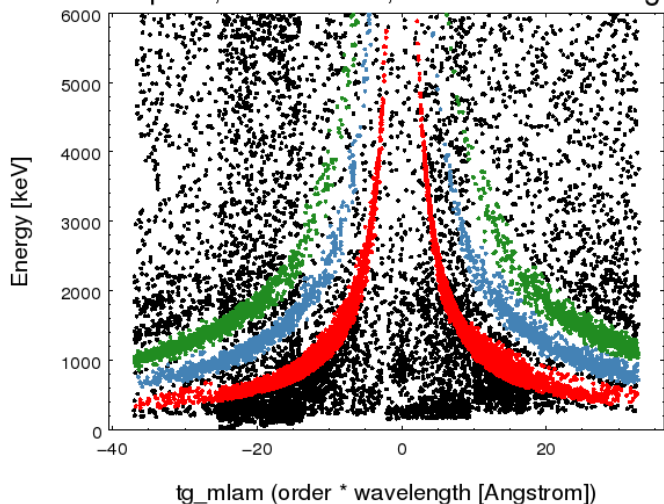


# Complexities in X-Ray and Chandra Data Analysis

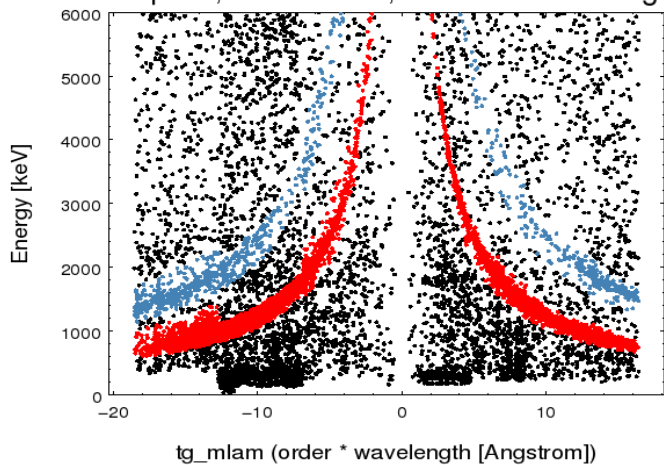
Capella: ACIS-S + HETG Raw Detector Image Color Coded



Capella, ObsId 9638, MEG Order Sorting



Capella, ObsId 9638, HEG Order Sorting



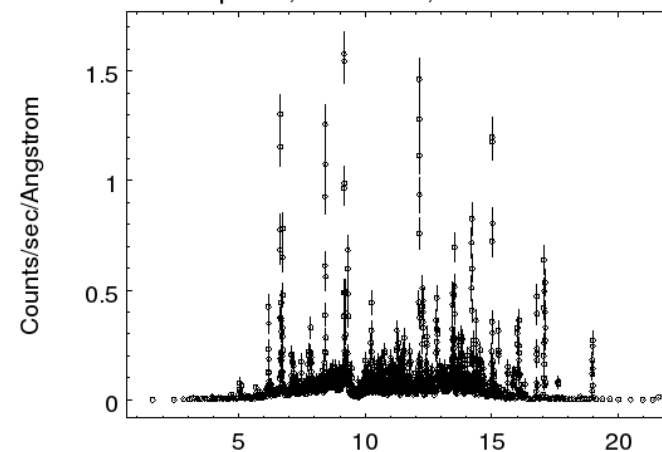
overlapping spectral orders  
separation

complicated geometry – need  
precise zero order location

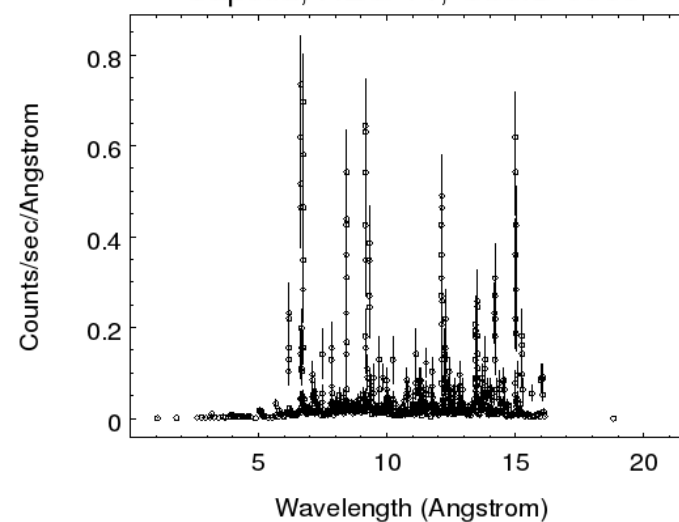
wavelength scale sensitive to  
details of chip geometry

background different on each  
chip

Capella, MEG +1, ObsID 9368



Capella, HEG +1, ObsID 9368



## Basics of CIAO

- Data files are in FITS format (usually binary tables, not just images)
- CIAO can also operate on ASCII file in many cases
- All (well, almost all) CIAO tools that want an input file can accept a CIAO Data Model “virtual file”
  - e.g instead of `evt.fits`
  - take `“evt.fits[energy=300:1000,sky=circle(4096,4096,20)]”`

Each file (dataset) is made up of sections called 'blocks' (HDUs for FITS fans)  
Blocks can be tables or images

Key tools:

```
dmcopy infile outfile  
dmlist infile opt=blocks,cols,data
```

```
ahelp dmlist → help for tool dmlist  
plist dmlist → list parameters for dmlist
```

Key applications:

```
Sherpa - fitting  
ChIPS – plotting  
ds9 – imaging and analysis
```

# The Event File

- In optical astronomy, the primary data set is an image. In radio interferometry, it's a visibility array.
- In X-ray astronomy, the primary data set is an event list - a table of (putative) photons
  - Our software makes it easy to generate an image from the event list, so it's easy to forget that's what you have. But making the image loses information.
  - First cut way of thinking about the event list: it's a 4-dimensional array of x, y, time, energy. But most pixels are empty (we don't have many photons!) so it's more compact to just list the non-empty ones.
  - Complication: we actually have many more parameters for each photon, not just 4.



# Inside the event list

```
jupiter> dmlist acisf03041_001N001_evt3.fits cols
```

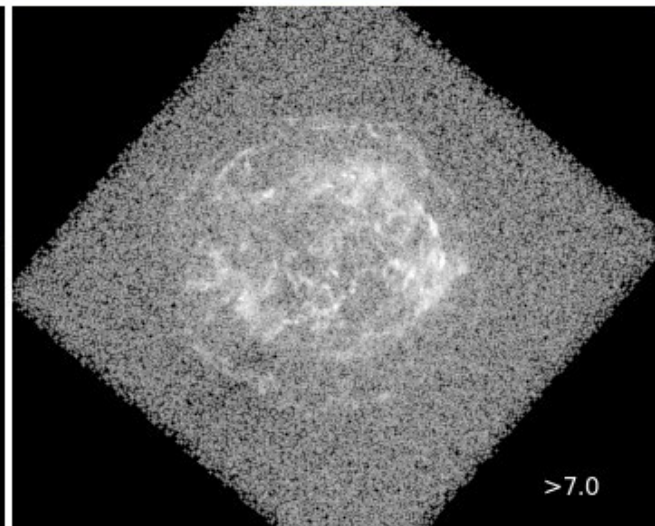
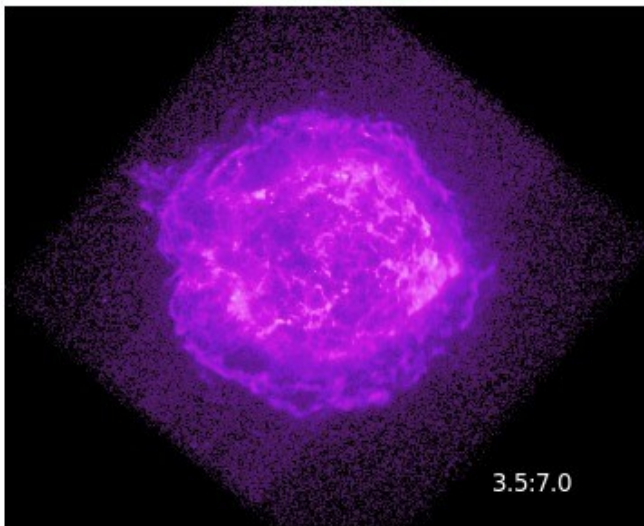
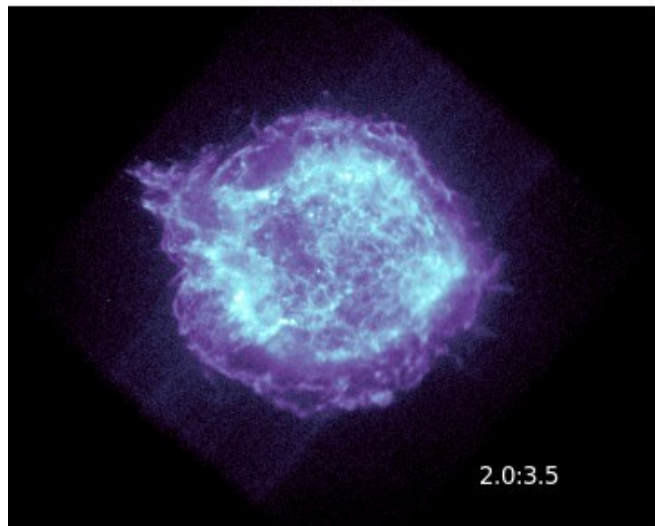
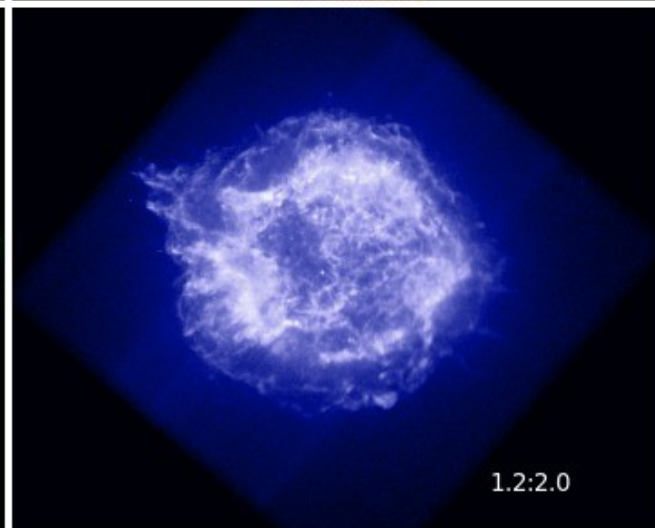
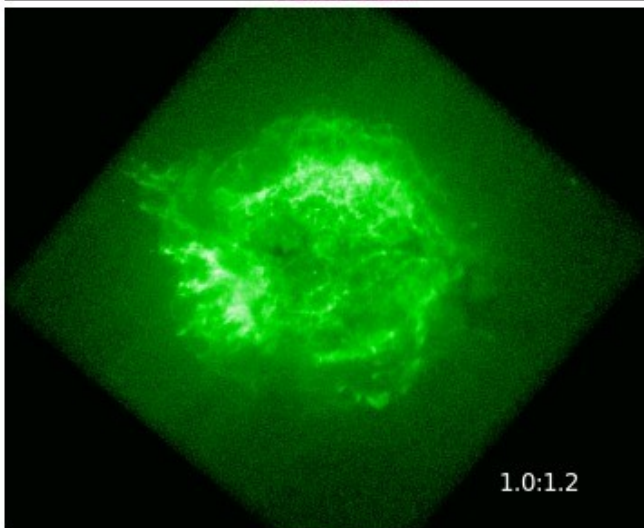
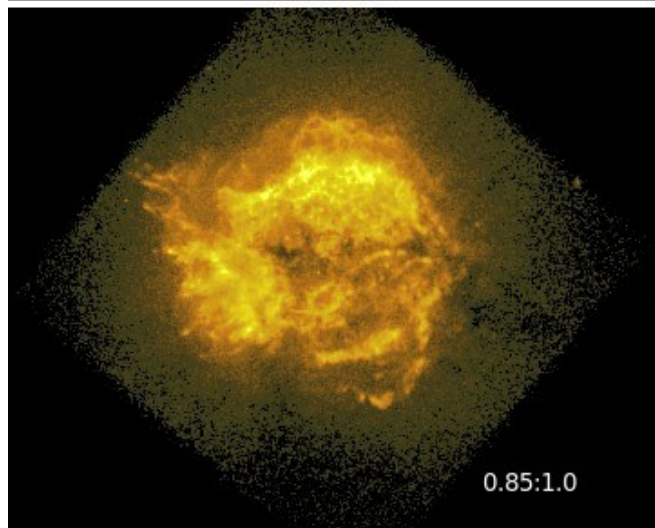
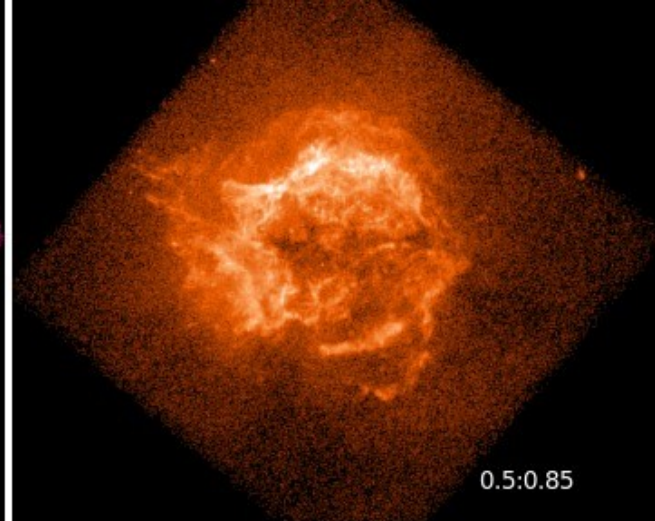
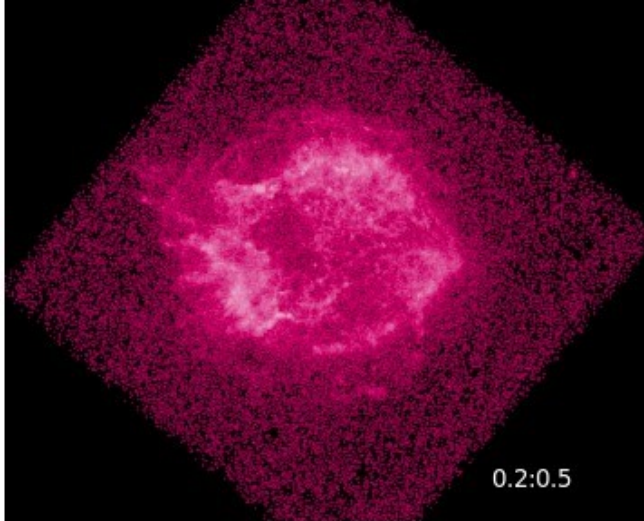
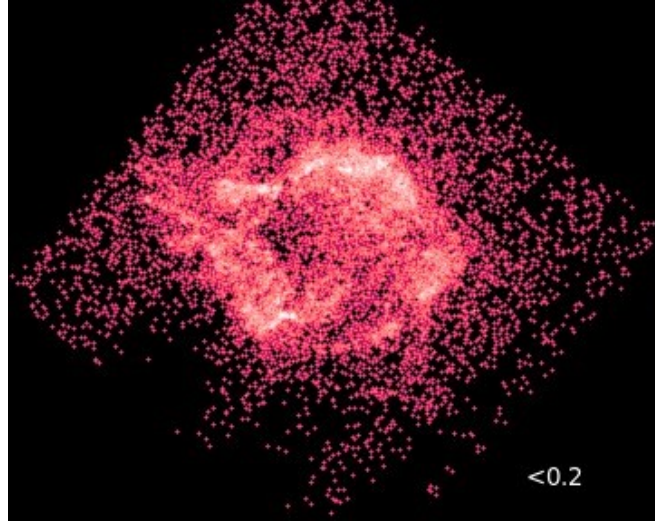
## Columns for Table Block EVENTS

ColNo	Name	Unit	Type	Range	
1	time	s	Real8	154361559.6127299964:154436827.4158599973	S/C TT corresponding to mid-exposure
2	ccd_id		Int2	0:9	CCD reporting event
3	node_id		Int2	0:3	CCD serial readout amplifier node
4	expno		Int4	0:2147483647	Exposure number of CCD frame containing event
5	chip(chipx,chipy)	pixel	Int2	1:1024	Chip coords
6	tdet(tdety,tdety)	pixel	Int2	1:8192	ACIS tiled detector coordinates
7	det(detx,dety)	pixel	Real4	0.50: 8192.50	ACIS detector coordinates
8	sky(x,y)	pixel	Real4	0.50: 8192.50	sky coordinates
9	pha	adu	Int4	0:36855	total pulse height of event
10	pha_ro	adu	Int4	0:36855	total read-out pulse height of event
11	energy	eV	Real4	0: 1000000.0	nominal energy of event (eV)
12	pi	chan	Int4	1:1024	pulse invariant energy of event
13	fltgrade		Int2	0:255	event grade, flight system
14	grade		Int2	0:7	binned event grade
15	status[4]		Bit(4)		event status bits

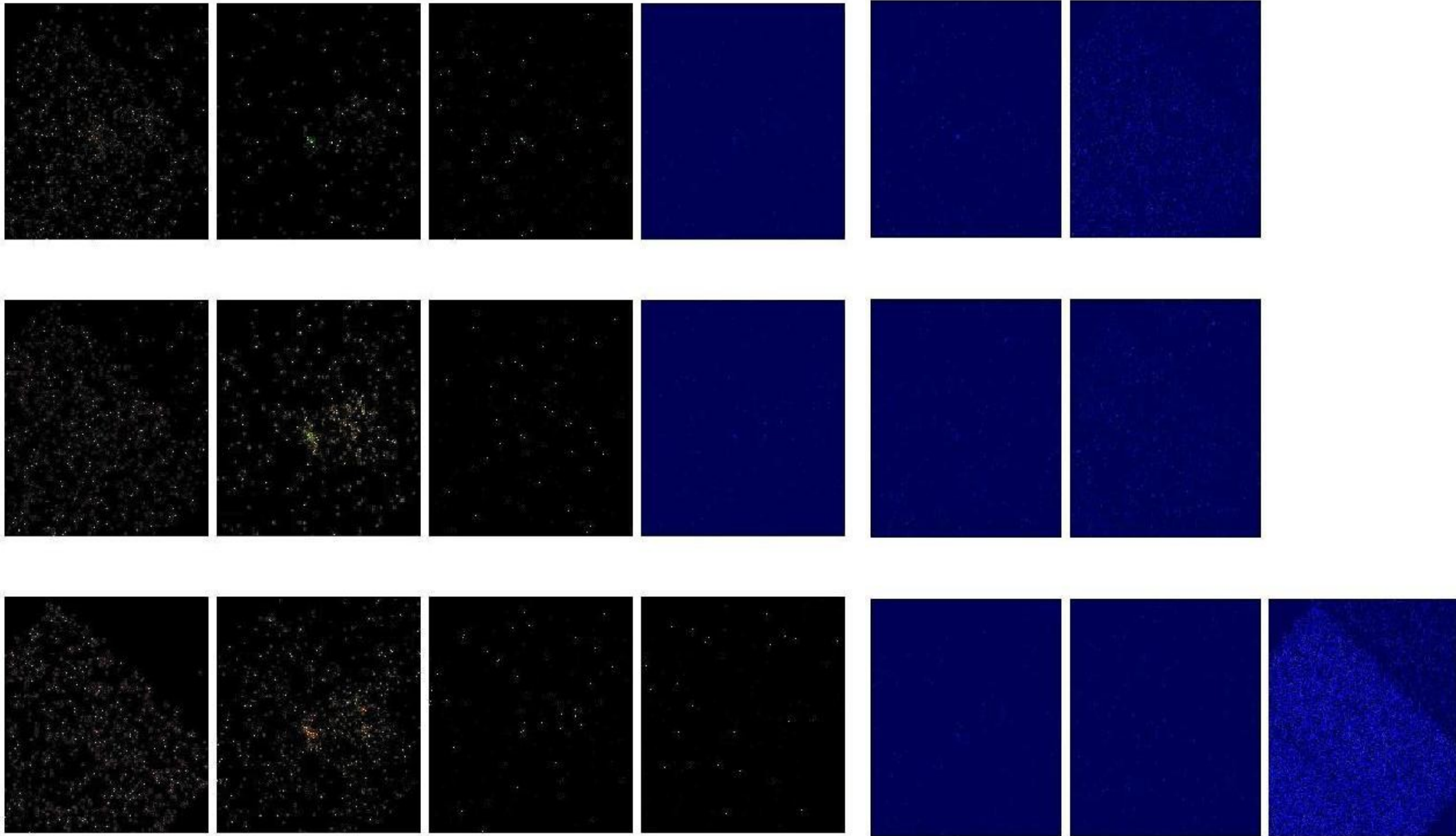
```
jupiter> dmlist acisf03041_001N001_evt3.fits["cols -status]" data,raw,clean rows=1:20
```

#	time	ccd_id	node_id	expno	chip(chipx,chipy)	tdet(tdety,tdety)	det(detx,dety)	sky(x,y)	pha	pha_ro	energy	pi	fltgrade	grade
154362662,7665936351	0	1	107	369 513	3574 4763	3540,2504882812	1556,8157958984	1473,2664794922	4129,1977539062	3688	3680	15358,6318359375	1024	16 4
154362662,7665936351	0	2	107	562 589	3650 4570	3615,2900390625	1748,9260253906	1676,4797363281	4093,9599609375	3977	3750	15254,2246093750	1024	64 2
154362662,7665936351	0	0	107	247 876	3937 4885	3902,1516113281	1435,5321044922	1426,5589599609	3750,3830566406	3765	3514	14473,1611328125	992	0 0
154362662,8076336384	7	0	107	189 301	4106 2003	4069,4038085938	4313,6518554688	4280,3339843750	4160,218750	3568	3503	15899,1279296875	1024	11 6
154362662,8076336384	7	1	107	264 388	4181 2090	4144,4223632812	4225,7763671875	4209,175781250	4069,1887207031	128	109	632,9125366211	44	72 6
154362662,8076336384	7	2	107	555 410	4472 2112	4435,0400390625	4204,0610351562	4245,8291015625	3780,0749511719	1717	1702	7969,5327148438	546	8 3
154362662,8076336384	7	2	107	676 441	4593 2143	4556,0268554688	4172,6386718750	4239,1547851562	3655,2526855469	1908	1853	8829,906250	605	16 4
154362662,8076336384	7	1	107	483 465	4400 2167	4363,3208007812	4149,5029296875	4178,0693359375	3839,4790039062	1011	994	4767,4877929688	327	2 2
154362662,8076336384	7	3	107	881 613	4798 2315	4760,8256835938	4001,3994140625	4112,1772460938	3420,4289550781	1348	1310	6024,1176757812	413	0 0
154362662,8076336384	7	2	107	690 834	4607 2536	4569,7802734375	3780,8713378906	3857,9919433594	3563,6794433594	2011	1942	9294,6806640625	637	72 6
154362662,8076336384	7	1	107	348 925	4265 2627	4228,7290039062	3689,4941406250	3700,4628906250	3879,6706542969	1722	1664	8069,8139648438	553	2 2
154362662,8076336384	7	1	107	502 954	4419 2656	4381,6430664062	3660,5825195312	3702,6135253906	3724,0622558594	3011	2957	14100,3310546875	966	208 6
154362662,8486736417	6	3	107	803 548	3678 2250	3639,6469726562	4066,6665039062	3952,636718750	4532,1162109375	2258	2095	8640,0263671875	592	2 2
154362662,8897136450	3	0	107	40 717	4415 3101	4376,0224609375	3215,3222666250	3265,1691894531	3640,8110351562	3146	2950	12326,56250	845	0 0
154362662,8897136450	3	0	107	78 901	4231 3139	4192,5224609375	3177,2634277344	3191,2951660156	3813,0412597656	2252	2077	8826,3281250	605	64 2
154362662,9307536483	2	0	107	208 240	3301 3878	3266,0554199219	2440,0964355469	2284,2392578125	4573,9658203125	3451	3423	13226,7988281250	906	16 4
154362662,9307536483	2	2	107	517 719	3780 3569	3744,9583984375	2748,1215820312	2681,5275878906	4166,1748046875	3816	3544	14376,746093750	985	64 2
154362662,9717936218	1	0	107	101 567	4565 4208	4527,886718750	2109,5788574219	2211,8908691406	3271,5732421875	1644	1543	6498,9438476562	446	64 2
154362662,9717936218	1	3	107	996 952	4180 5103	4141,839843750	1216,6264648438	1259,9039306641	3471,8693847656	2406	2170	9509,4218750	652	16 4
154362666,0075938106	0	2	108	683 156	3217 4449	3183,1049804688	1869,4627685547	1709,3774414062	4541,6933593750	3466	3429	13290,0634765625	911	0 0



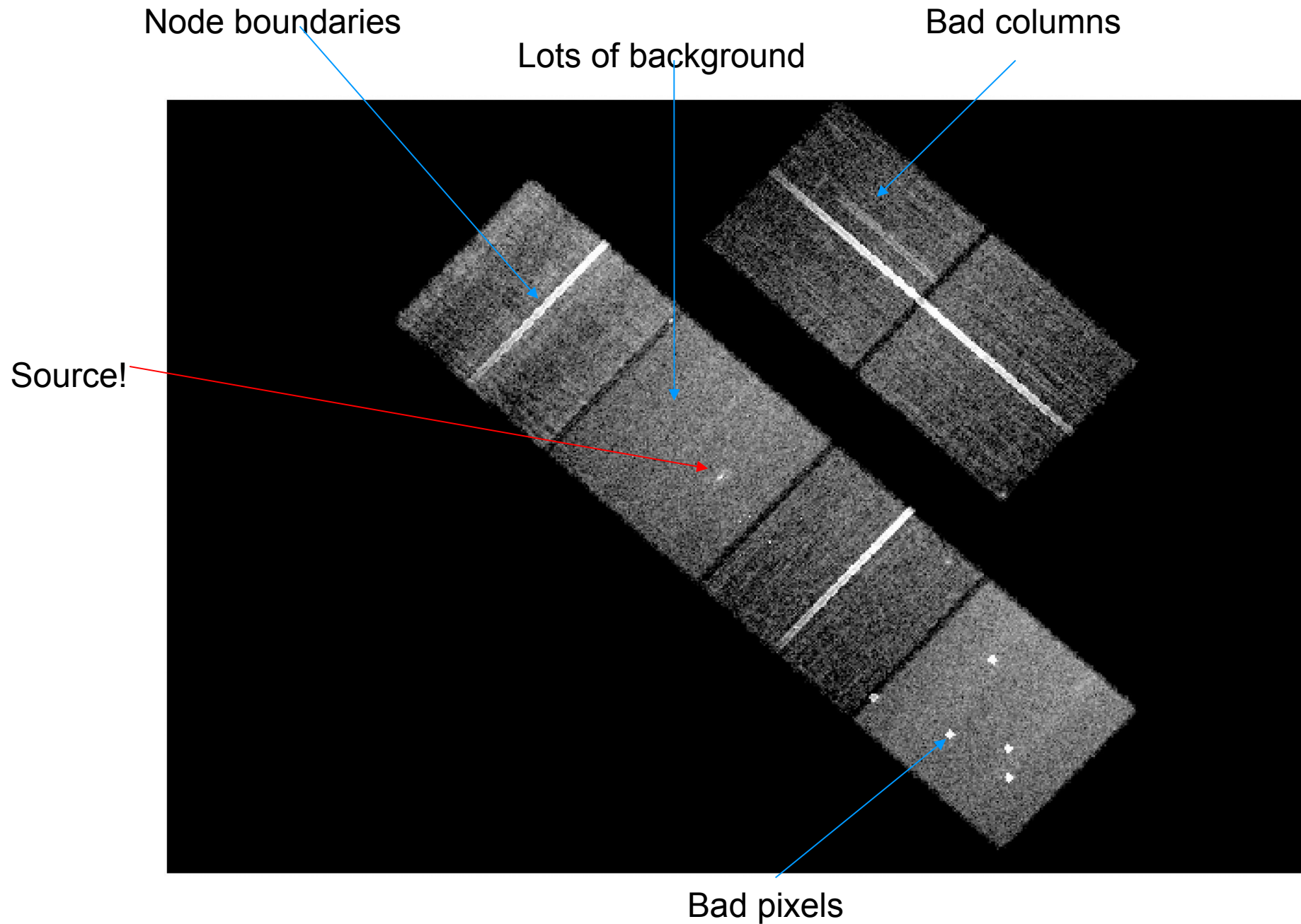


Energy slices through an event list, 0.1 - 10 keV





# Level 1 Event List - Calibrated but Dirty



# Level 2 event list - cleaned and filtered

Energy filter 300-7000 eV removes background but not signal

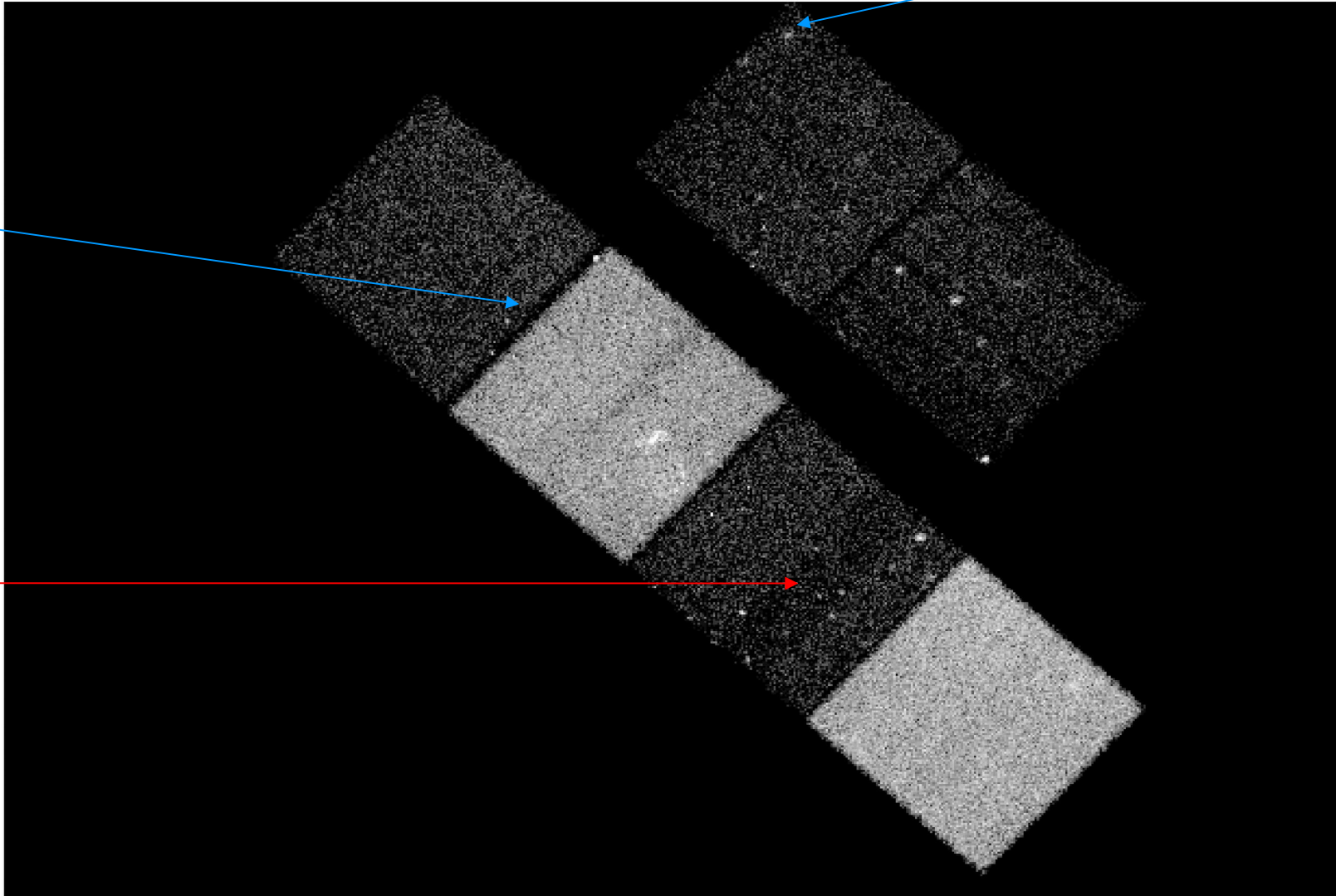
Grade filter removes cosmic ray events etc

Good time filter removes times of high background, poor data quality

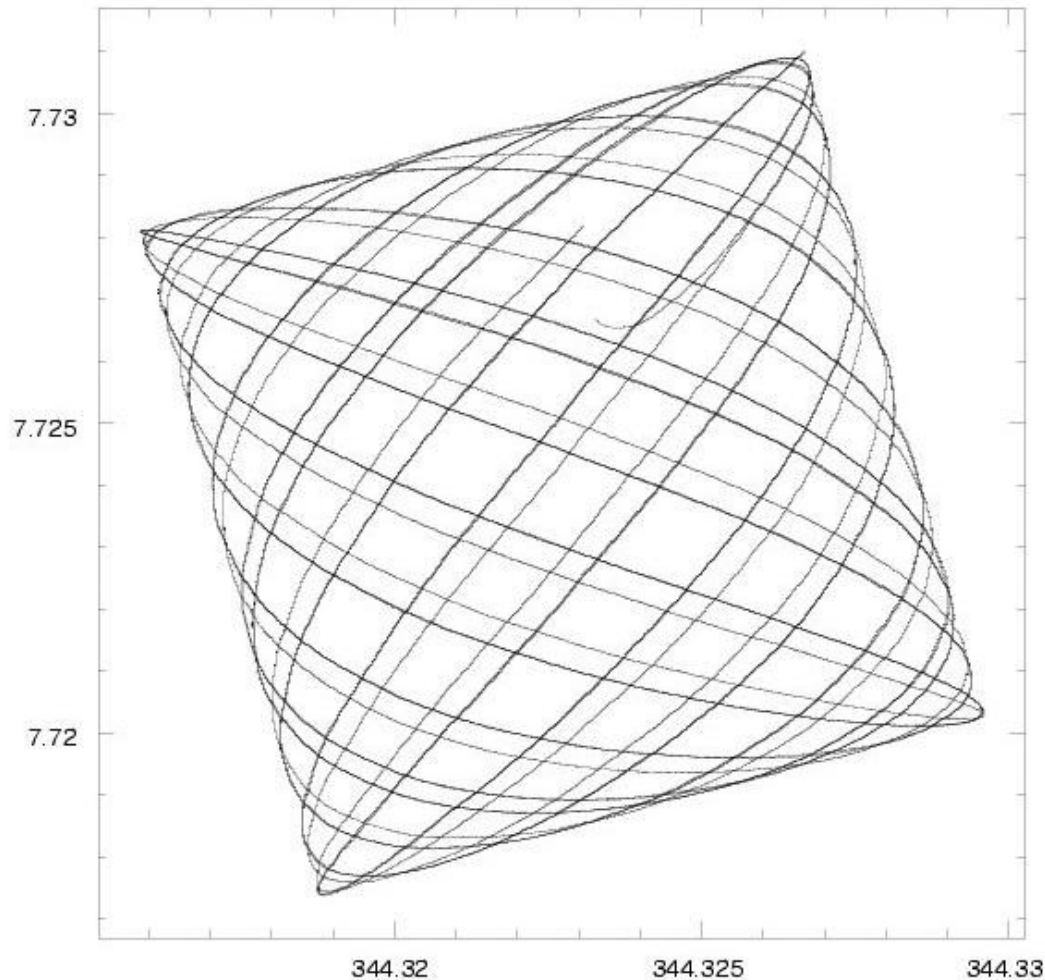
Sources fuzzy far  
off axis (PSF big)

Beware  
chip gaps!

More  
sources!



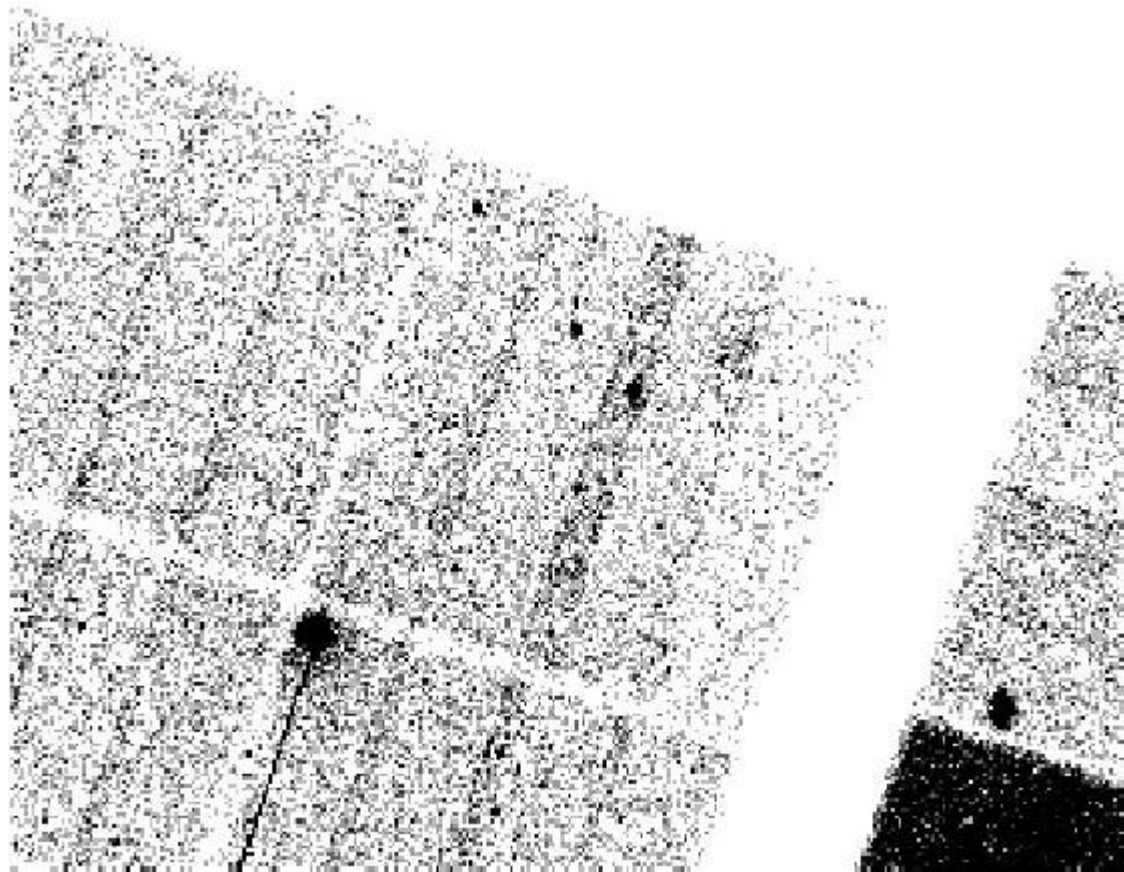
# The aspect solution



During an observation, Chandra's optical axis describes this 'dither pattern' on the sky, (Problem 5), smearing the image of a point source. The RA, Dec, roll angle of the telescope versus time is called the 'aspect solution'; the asol1.fits file provides this for each observation.

We record the motion of the guide stars in the star tracker so that we can calculate RA and Dec for EACH PHOTON and so reconstruct the image.

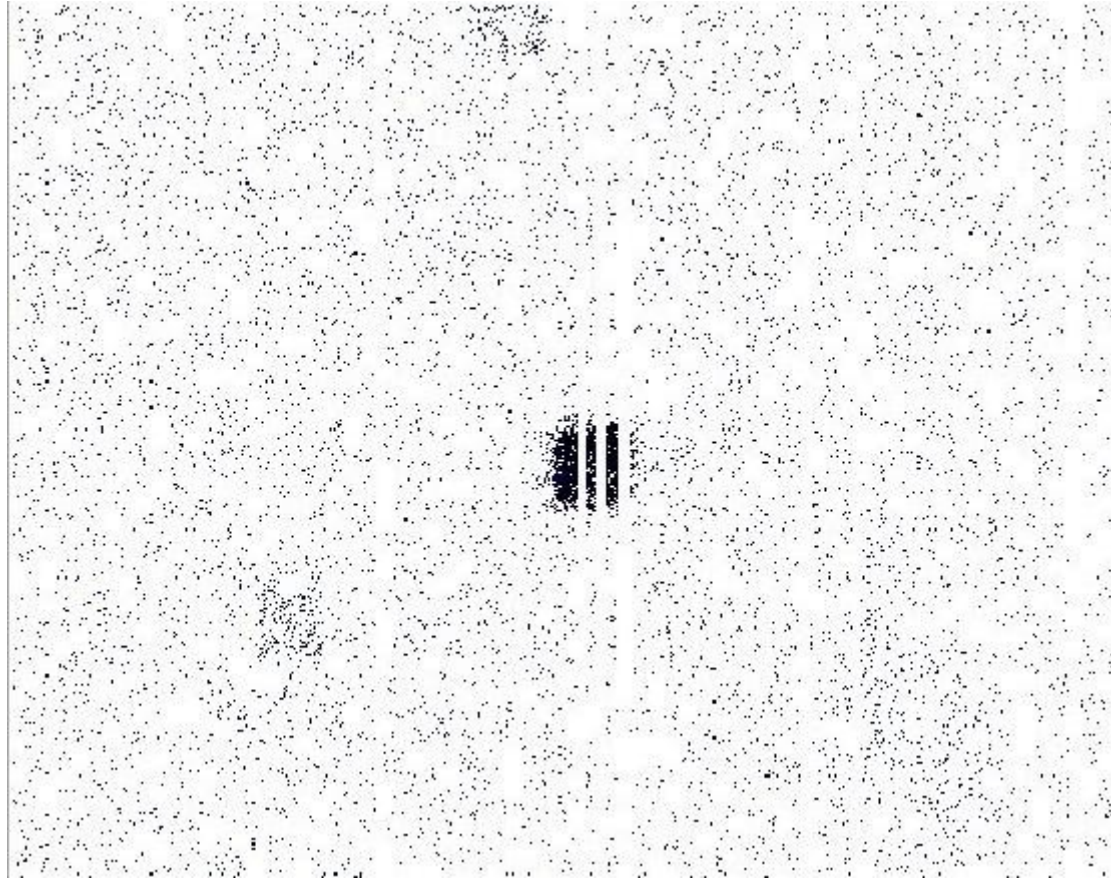
# Chandra aspect-corrected data



This is what you get after calibration but before cleaning the data. Note the sharp point sources near the center.



## Chandra raw (chip) data



In instrument space, the photons are spread out over 20 arcsec and have bad columns going through them - so be careful of the effective exposure time. If you didn't dither, you could lose the source entirely if it landed on a bad pixel

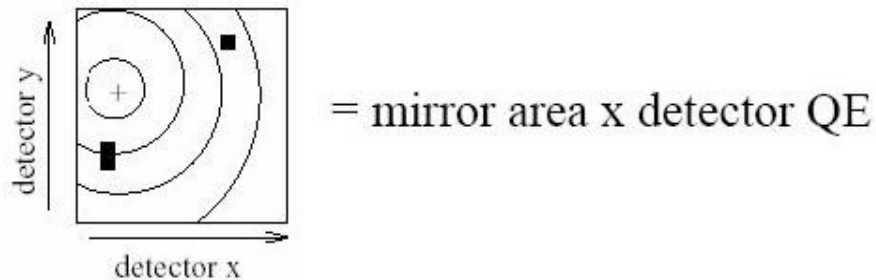
## Spatial Response: EXPOSURE MAP

The *Exposure Map*,  $E(\Delta h, \lambda, \hat{p})$  retains spatial information at the expense of spectral. It has units of  $[\text{cm}^2 \text{ counts photons}^{-1}]$ .

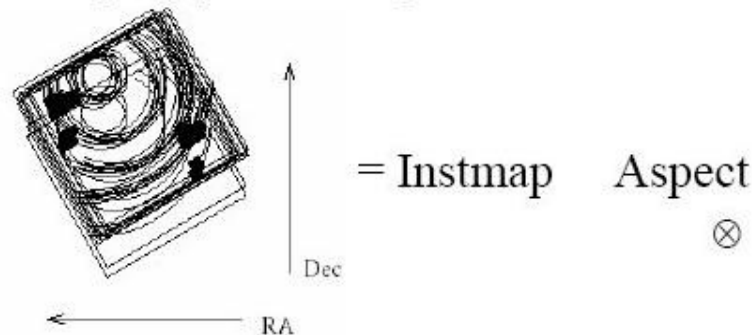
$$\int d\lambda S(\lambda, \hat{p}) \approx \frac{C(\Delta h, \hat{p})}{E(\Delta h, \lambda, \hat{p})}$$

$C$  is the observed counts per spatial bin in a pulse-height bin.  $S$  is the source flux, with units of  $[\text{phot cm}^{-2}\text{s}^{-1}\text{\AA}^{-1}]$ .

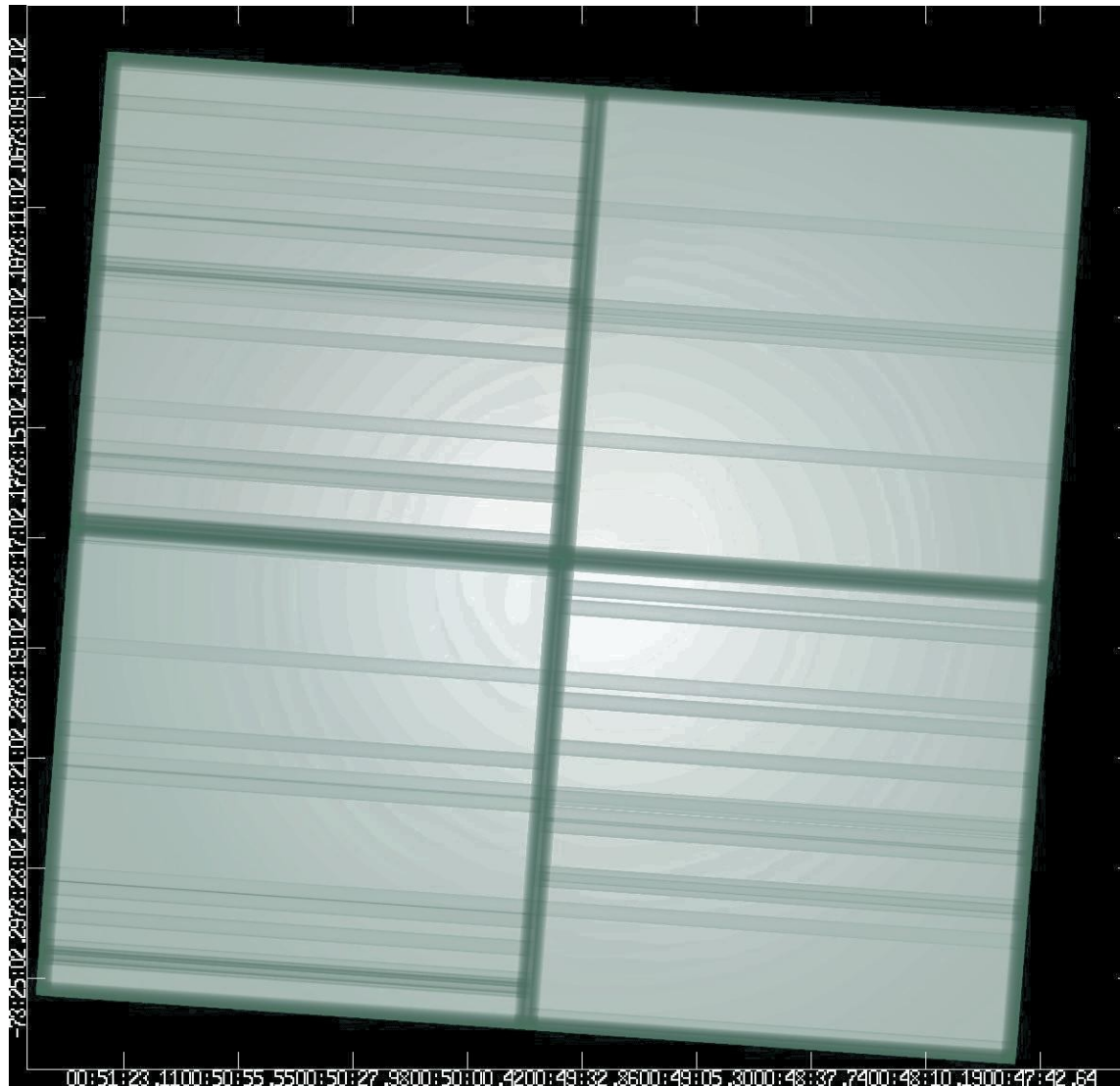
**Instrument Map** – efficiency calibration information, band integrated. (create with `mkinstmap`)



**Exposure Map** – applies telescope aspect history and coordinate transformations (= area x time). (create with `mkexpmap`).

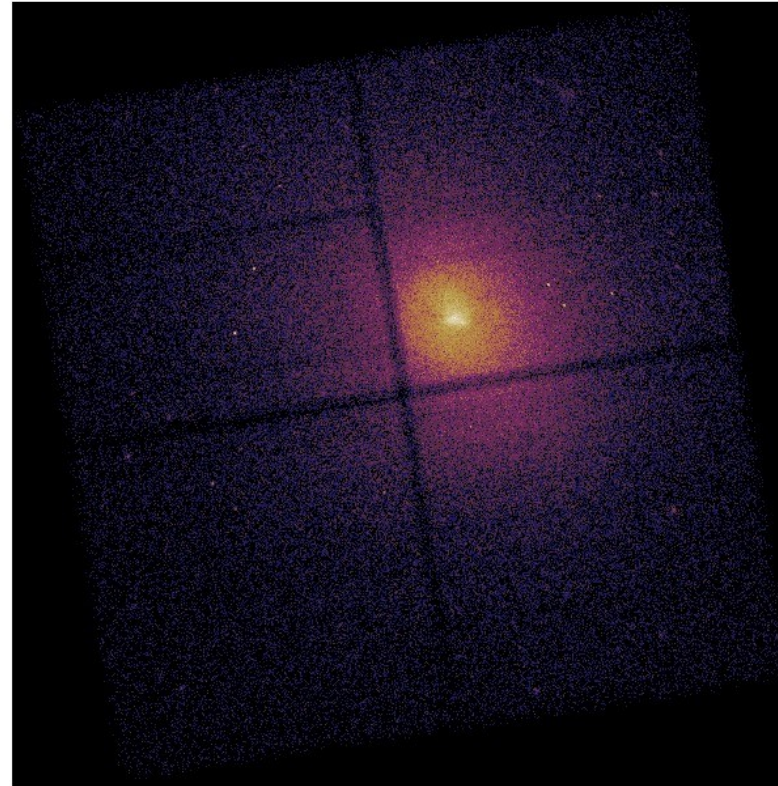
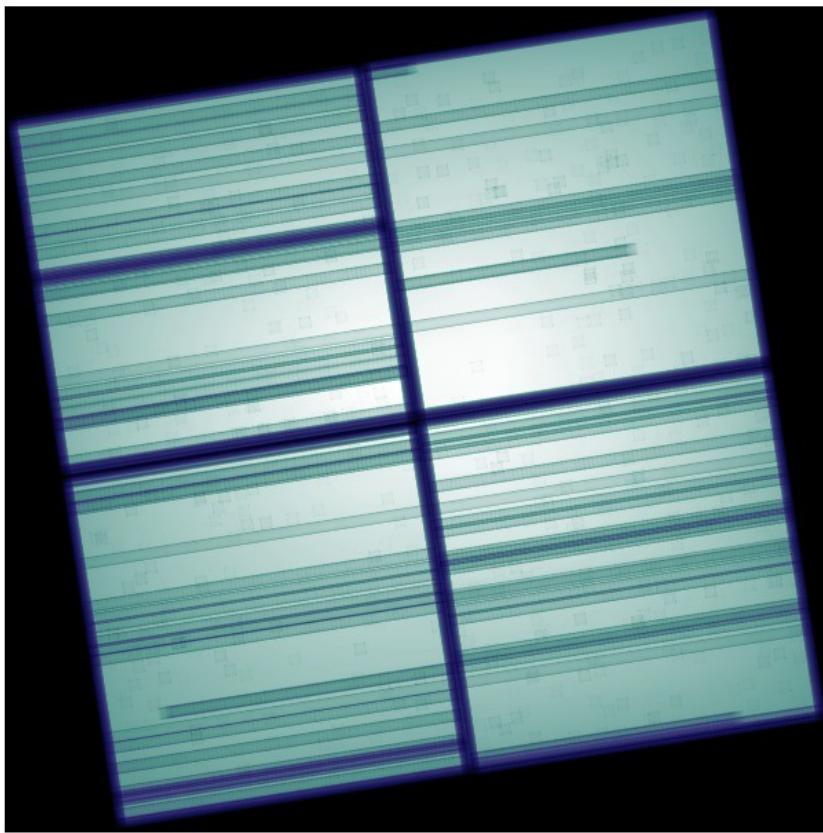


# Typical exposure map

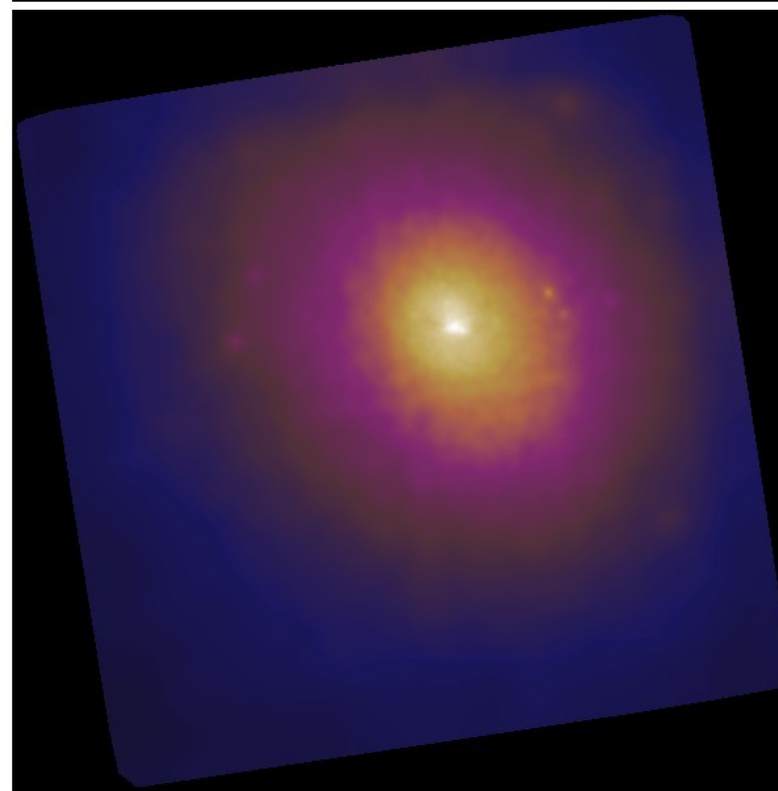


**Problem 3:** Exposure map is energy dependent; must assume a spectrum if using a broad band





Applying exposure map  
(left) to counts image (top  
right) and then smoothing  
(bottom right) reveals more  
structure



## Event analysis or binned analysis?

- Don't make an image too quickly. If you can get an answer directly from the event list, that's better - binning the data loses information, and collapsing the axes loses information.
- Spatial analysis: make an image (using `dmcopy`)
  - lose energy and time information
- Spectral analysis: make a 'PHA file' using `dmextract` (or a grating spectrum using `tgextract` )
  - lose spatial and time information
- Temporal analysis: make a light curve using `dmextract`

# The fundamental equation of astronomy

$$N(E) = A(E)F(E)\Delta T$$

Our instrument makes a spectrophotometric measurement; the sensitivity (“effective area”)  $A(E)$  tells us how to convert from flux to instrumental counts for a given exposure time  $\Delta T$

But, a real instrument doesn't measure the true energy, it measures instrumental energy  $E'$ . The line spread function (“response matrix” in X-rays)  $R(E, E')$  describes how a monochromatic input spectrum is broadened by the instrument (**Problem 2**)

Let us further assume that the instrumental energy  $E'$  is measured in discrete channels (bins)  $E'_i$ . Then

$$N(E'_i) = \int A(E)R(E, E'_i)F(E)dE\Delta T$$

Of course, you may not be measuring all of the light from the source. Even if it's a point source, there may be an aperture correction. We need the PSF  $P(x-x', y-y')$  and the spatial dependence of the QE,  $q(x, y)$ . Then at a given instrument position  $x', y'$

$$N(E'_i, x'_i, y'_i) = \int \int A(E)R(E, E'_i)F(E, x, y)P(x-x'_i, y-y'_i)q(E, x'_i, y'_i)dEdxdy\Delta T$$

The source may also be variable in time - we'll ignore this for the purposes of this talk. The detector sensitivity is time-variable on long timescales, but for a single observation you just have to worry about times when the data is filtered - the Good Time Intervals (GTIs)

$$N(E'_i, x'_i, y'_i) = \int \int \int A(E)R(E, E'_i)F(E, x, y, t)P(x-x'_i, y-y'_i)q(E, x'_i, y'_i)dEdxdydt$$



# Pulse height

When you plot an optical spectrum, the wavelength (or energy) axis is really an instrumental quantity. A spectral line is broadened by instrumental effects, so the energies plotted are not the true energies of the photon. However, the instrument is calibrated (i.e. the definition of instrumental energy is rescaled) such that the peak of a line is at the correct energy.

In X-ray astronomy, instead of using the instrumental energy  $E'$ , we work with the energy bin number. For historical reasons to do with long-forgotten instruments, this bin number is known as the PI channel (for 'pulse invariant' channel) - we'll denote it by  $P$ . So, for fixed energy bin widths  $dE$ ,

$$E' = P dE = [\text{on average}] E$$

The instrument actually measures a raw energy bin number  $p$ , called the PHA channel, or 'pulse height analyser channel'. The scaling of the instrumental energy to real energy depends on position and time:

$$E'(\text{raw}) = p dE = g(x,y,t)P dE$$

This function  $g$  (the gain) is usually assumed to obey

$$g(x,y,t) = g_{\text{spatial}}(x,y) g_{\text{t}}(t)$$

and we provide calibrations of both the spatial gain and the temporal gain.

# Spectra in Poissonland

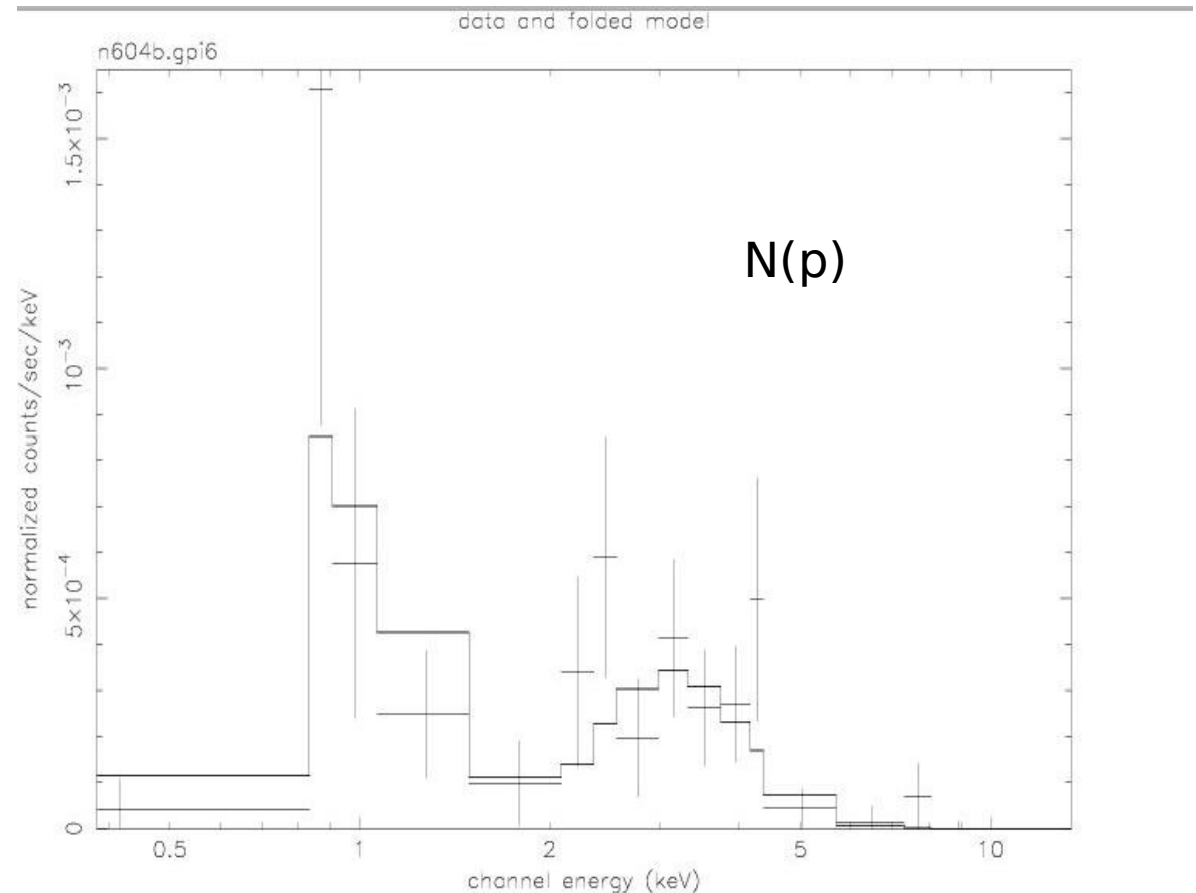
We pick a parameterized  $F(E)$  such as warm absorber models, lines, thermal plasma codes. Which  $F(E)$ ? You must pick one based on expected physics, but match number of free parameters with quality of data.

With less than 100 counts, we usually just use count ratios (X-ray colors) for spectral analysis.

Does one model fit significantly better than another? Be careful that two physically different models may look quite similar in  $F(E)$  space.

Incompletely calibrated instrumental features may show up in residuals, limiting factor in high S/N spectra - these features may include edges. Beware apparent science in regions where  $A(E)$  is changing rapidly.

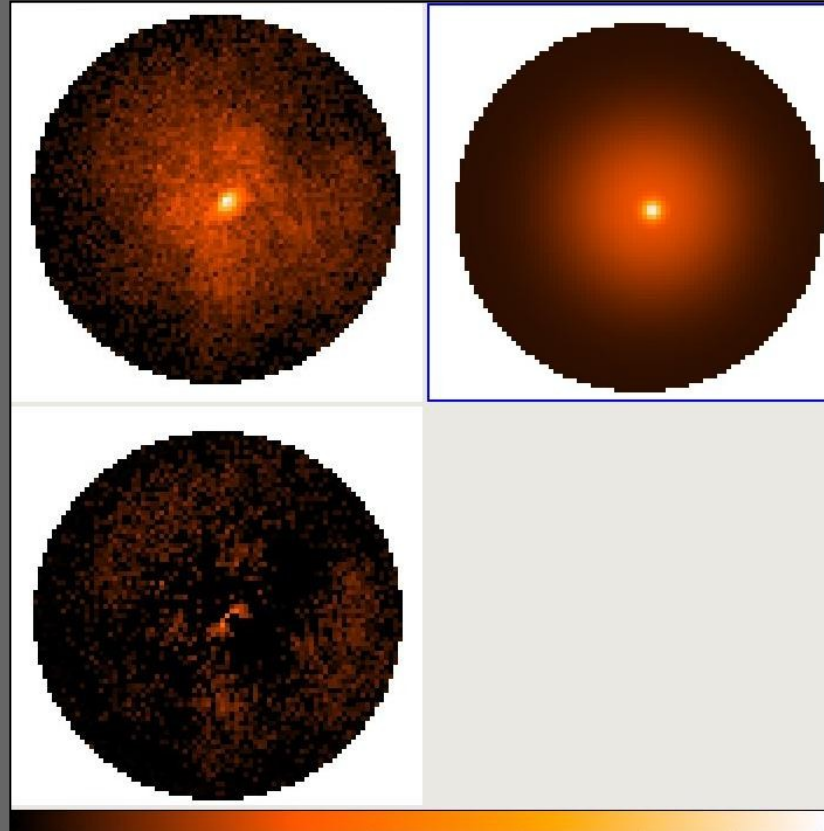
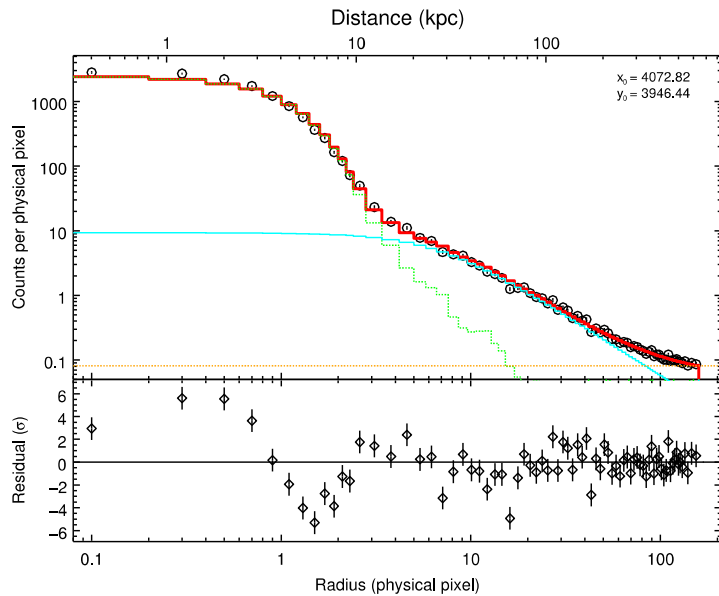
$$N(p) = \int R(E, p) A(E) F(E) dE$$



# Sherpa: Modeling and Fitting in Python



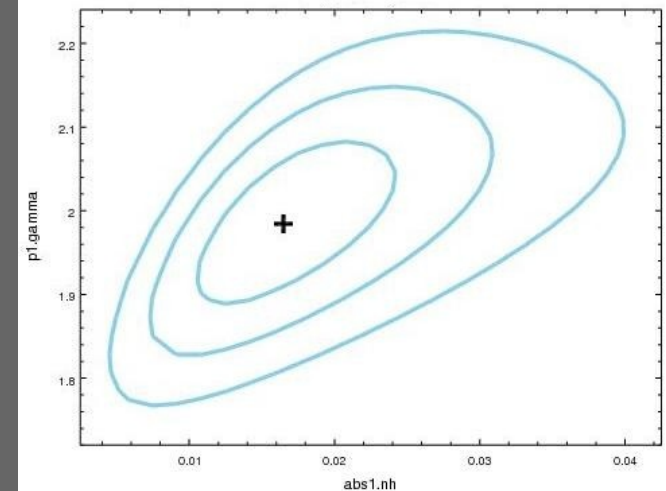
Modeling and fitting for 1-D and 2-D datasets **in any waveband** including: spectra, images, surface brightness profiles, light curves, general ASCII data.



Coded in a Python environment – familiar to the new generation of astronomers and used in other missions

Model Poisson and Gaussian data

Calculate confidence levels on the best-fit model parameters

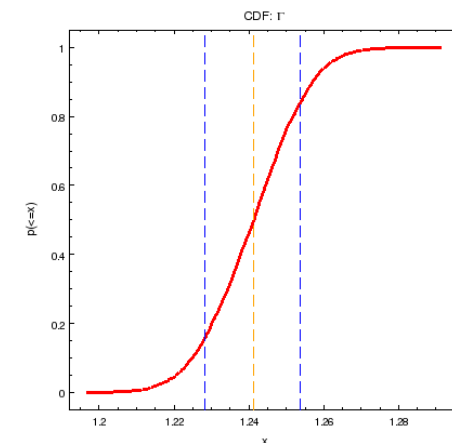
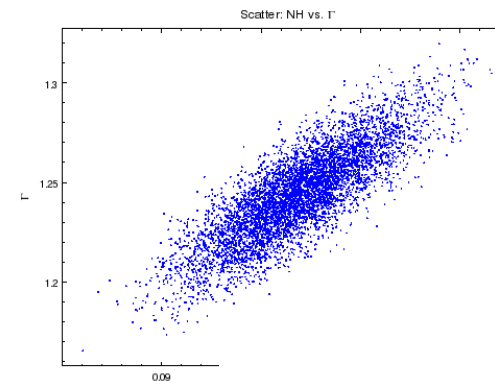
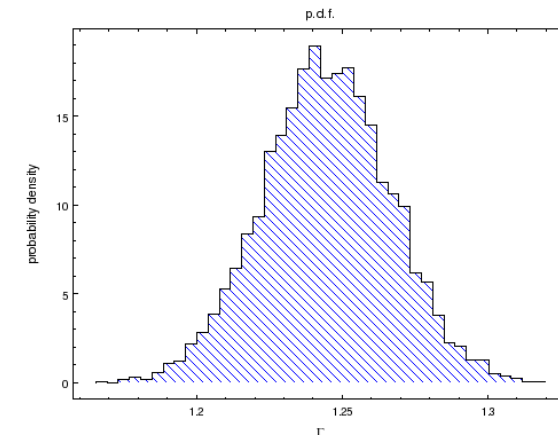


# Sherpa: Modeling and Fitting in Python



Sherpa

- comes with well-tested, robust optimization methods - e.g. Levenberg-Marquardt, Nelder-Mead Simplex or Monte Carlo/Differential Evolution
- comes with statistics for modeling Poisson or Gaussian data
- can perform Bayesian analysis with Poisson Likelihood and priors, using Metropolis or Metropolis-Hastings algorithm in the MCMC (Markov-Chain Monte Carlo); allows to include non-linear systematic errors (calibration uncertainties) in the analysis
- is extensible (with python and compiled code):
  - is used in CIAO tools and scripts
  - in the Xija Chandra thermal modeling code
  - is used in the TeV HESS data analysis software
  - is used in the IRIS spectral energy distribution program



# CIAO scripts: analysis simplified

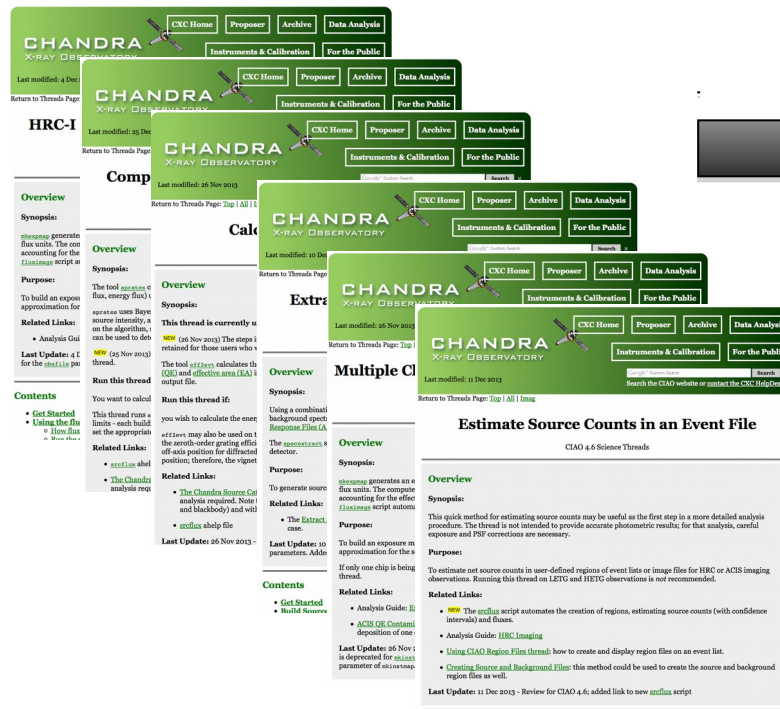
Recent emphasis on high level programs with easy interfaces – particularly helpful for users who are not X-ray astronomy specialists.

Wrap laborious thread analysis steps with a single command line script

Handle the various special cases by inspecting the metadata in the data files

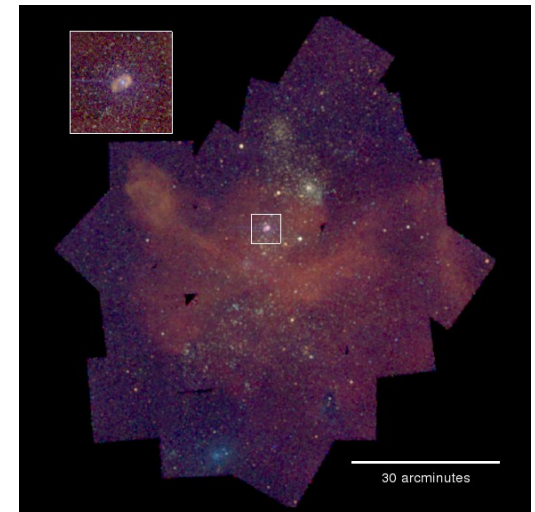
Scripts have parameters which allow the expert user to tune them

This new suite of scripts makes analysis quicker



one script

Two examples: srcflux and merge\_obs





## MAKING X-RAY ANALYSIS EASIER

- `ciao_install` - automated installation process
- tunable, also supports source builds

What data is there? WebChaser is still great, but sometimes `find_chandra_obsid` is handy for CL use or scripting::

```
neptune>
neptune> find_chandra_obsid "NGC 2403"
# obsid  sepn  inst grat  time  obsdate  piname  target
2014     0.2 ACIS-S NONE  36.0 2001-04-17  CAPPI  NGC2403
2937     2.6 HRC-I NONE   2.8 2002-01-27  SUGIHO "NGC2403 S3"
4628     2.7 ACIS-S NONE  47.1 2004-08-23  Lewin  "SN 2004dj"
4629     2.7 ACIS-S NONE  45.1 2004-10-03  Lewin  "SN 2004dj"
4630     2.7 ACIS-S NONE  50.6 2004-12-22  Lewin  "SN 2004dj"
neptune> □
```

There's also the footprint service [cxc.harvard.edu/cda/footprint](http://cxc.harvard.edu/cda/footprint)

`find_chandra_obsid` can also download the data, or you can use...



```

neptune>
neptune>
neptune> download_chandra_obsid 4628,4629
Downloading files for ObsId 4628, total size is 204 Mb.

```

Type	Format	Size	0.....H.....1	Download Time	Average Rate
oif	fits	25 Kb	#####	< 1 s	179.5 kb/s
vv	pdf	40 Kb	#####	< 1 s	224.1 kb/s
full_img	jpg	60 Kb	#####	< 1 s	422.4 kb/s
cntr_img	fits	175 Kb	#####	< 1 s	933.2 kb/s
full_img	fits	92 Kb	#####	< 1 s	603.0 kb/s
evt2	fits	21 Mb	#####	7 s	2882.5 kb/s
asol	fits	12 Mb	#####	4 s	2715.4 kb/s
bpix	fits	44 Kb	#####	< 1 s	385.3 kb/s
fov	fits	6 Kb	#####	< 1 s	68.0 kb/s
eph1	fits	282 Kb	#####	< 1 s	1150.0 kb/s
cntr_img	jpg	573 Kb	#####	< 1 s	1604.3 kb/s
stat	fits	2 Mb	#####	< 1 s	2201.2 kb/s
flt	fits	6 Kb	#####	< 1 s	67.6 kb/s
msk	fits	5 Kb	#####	< 1 s	65.2 kb/s
mtl	fits	2 Mb	#####	< 1 s	2504.2 kb/s
evt1	fits	125 Mb	#####	53 s	2440.5 kb/s
bias	fits	443 Kb	#####	< 1 s	1198.8 kb/s
bias	fits	493 Kb	#####	< 1 s	1394.2 kb/s
bias	fits	448 Kb	#####	< 1 s	1495.9 kb/s
bias	fits	431 Kb	#####	< 1 s	1373.8 kb/s
bias	fits	431 Kb	#####	< 1 s	1313.7 kb/s
bias	fits	441 Kb	#####	< 1 s	1300.8 kb/s
pbk	fits	4 Kb	#####	< 1 s	45.1 kb/s
vv	pdf	35 Mb	#####	16 s	2255.8 kb/s

`download_chandra_obsid` gets the data for you  
This one makes subdirs 4628/ and 4629/ each with the usual  
primary/, secondary/ subdirs that you are used to

Next we update the archive processing with the latest calibrations using `chandra_repro`

```
neptune> ls 4628
axaff04628N002_VV001_vv2.pdf  oif.fits  primary/  secondary/
neptune> chandra_repro
Input directory (./): 4628
Output directory (default = $indir/repro) ():
```

Now we have a new `repro/` subdirectory with (hopefully) all the files you'll need for further analysis, including “`repro_evt2.fits`”

```
neptune> ls 4628
axaff04628N002_VV001_vv2.pdf  oif.fits  primary/  repro/  secondary/
neptune> ls 4628/repro
acisf04628_000N003_bpix1.fits  acisf04628_000N003_stat1.fits  acisf209642202N003_pbk0.fits
acisf04628_000N003_fov1.fits  acisf04628_asol1.lis          pcadf209643885N003_asol1.fits
acisf04628_000N003_msk1.fits  acisf04628_repro_bpix1.fits
acisf04628_000N003_mt11.fits  acisf04628_repro_evt2.fits
neptune> □
```

`chandra_repro` also works on grating data



Now you have calibrated data and are ready to do science.

You may want to take a look at the data by making a three color fluxed image using 'fluximage'; cd into the repro directory and run as shown here.

- knows about CSC bands soft, med, hard, broad

- finds the asol, badpix, mask etc. on its own

- makes exposure maps etc.:

```
neptune> fluximage *repro_evt2.fits out=fimg bin=4 bands=CSC
Running fluximage
Version: 08 November 2012
```

```
Using CSC ACIS soft science energy band.
Using CSC ACIS medium science energy band.
Using CSC ACIS hard science energy band.
Aspect solution pcadf209643885N003_asol1.fits found.
Bad pixel file acisf04628_repro_bpix1.fits found.
Mask file acisf04628_000N003_msk1.fits found.
PBK file acisf209642202N003_pbk0.fits found.
```

```
The output images will have 1301 by 1286 pixels, pixel size of 1.968 arcsec,
and cover x=1336.5:6540.5:4,y=1672.5:6816.5:4.
```

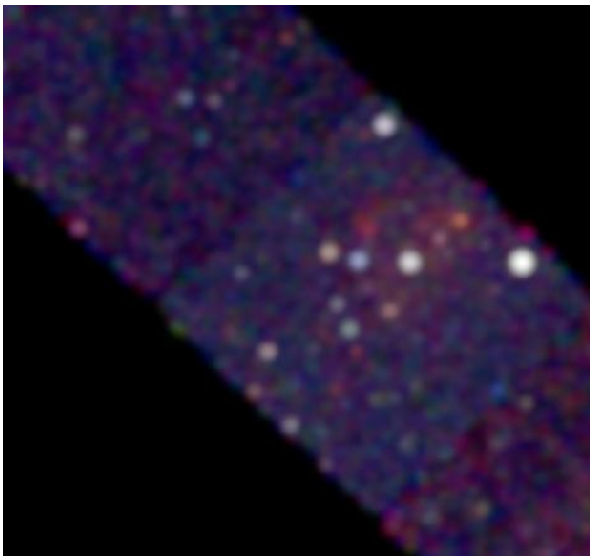
```
Running tasks in parallel with 8 processors.
Creating aspect histograms for obsid 4628
Creating 18 instrument maps for obsid 4628
Creating 18 exposure maps for obsid 4628
Combining 6 exposure maps for 3 bands (obsid 4628)
Thresholding data for obsid 4628
Exposure-correcting 3 images for obsid
```

The following files were created:

```
The clipped counts images are:
fimg_soft_thresh.img
fimg_medium_thresh.img
fimg_hard_thresh.img
```

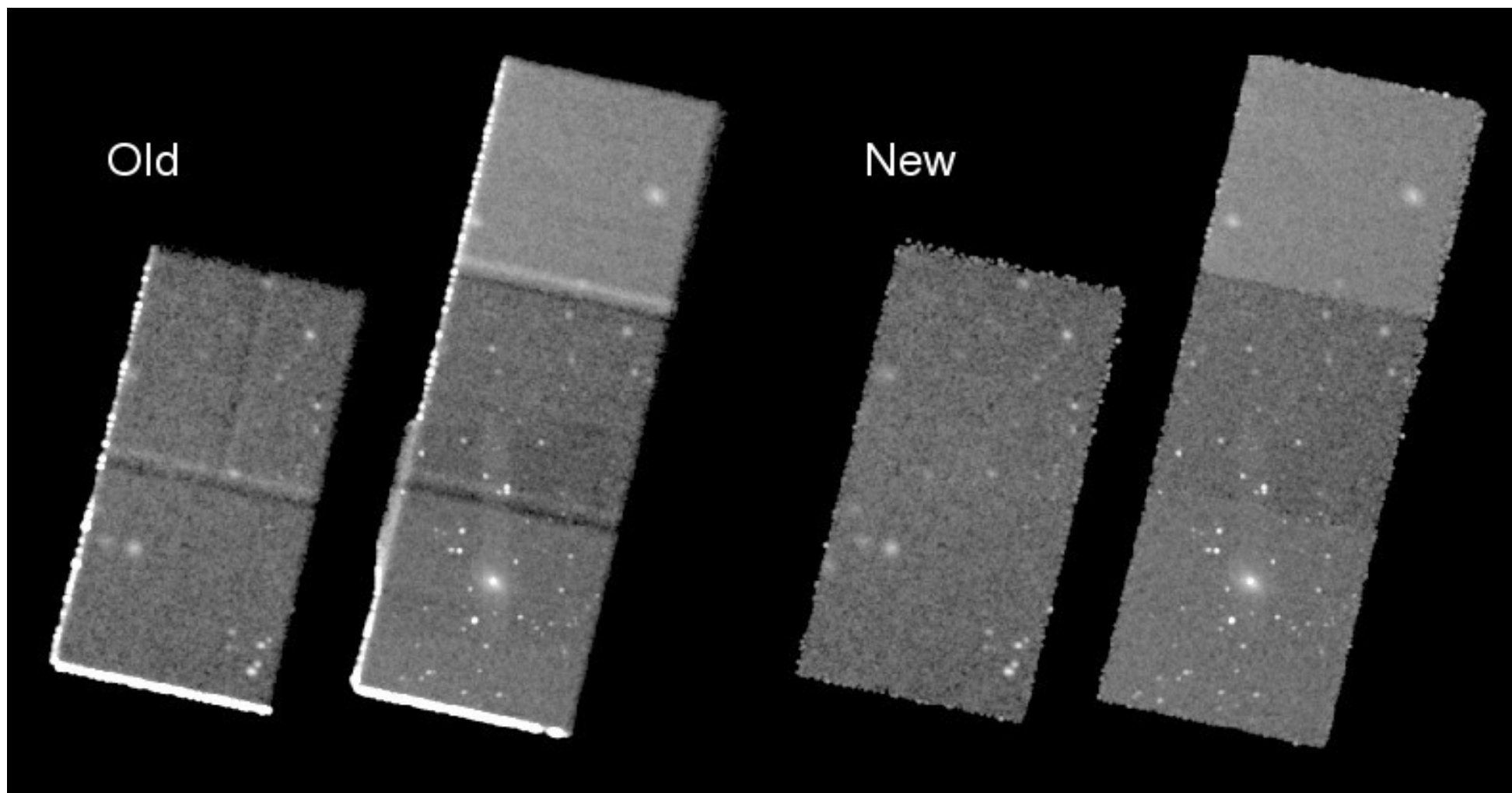
```
The clipped exposure maps are:
fimg_soft_thresh.expmmap
fimg_medium_thresh.expmmap
fimg_hard_thresh.expmmap
```

```
The exposure-corrected images are:
fimg_soft_flux.img
fimg_medium_flux.img
fimg_hard_flux.img
```





## Combining Observations – Example 1



Adding four observations shows limitations of old script: obsid no 3 has a different SIM position and obsid 4 is a subarray; the new script handles the exposure maps and reprojection correctly in these cases. Avoid bad pixels at edge with thresholding



## merge\_obs – Summary.

---

### The new script

- parallelizes the computation across multiple processors on the host machine
- automatically determines the center and size of the mosaic (if the user doesn't specify) by averaging the unit vectors of the pointing directions and taking the union of the projected field-of-view polygons
- modifies headers to account for the fact that the 'sky' pixel coords go beyond their normal range (which can cause ds9 not to display part of the image)
- automatically handles different event input formats by trimming columns as needed
- automatic location and use of mask, aspect, bad pixel, parameter block files using values seeded in event file header
- sorts input files in time order
- for HRC-I, subtract particle background model
- thresholds final image using exposure map (default 1.5% of max exposure)
- cleans up intermediate files on exit
- supports standard catalog energy bands e.g. 'CSC', 'soft' as well as user-specified ones; can use spectral weight files for exposure maps if supplied

### Limitations:

- Cannot combine ACIS with HRC-I/S, or HRC-I with HRC-S
- No ACIS background subtraction
- No support yet for improving astrometry before merging





## Combining Observations – Example 2



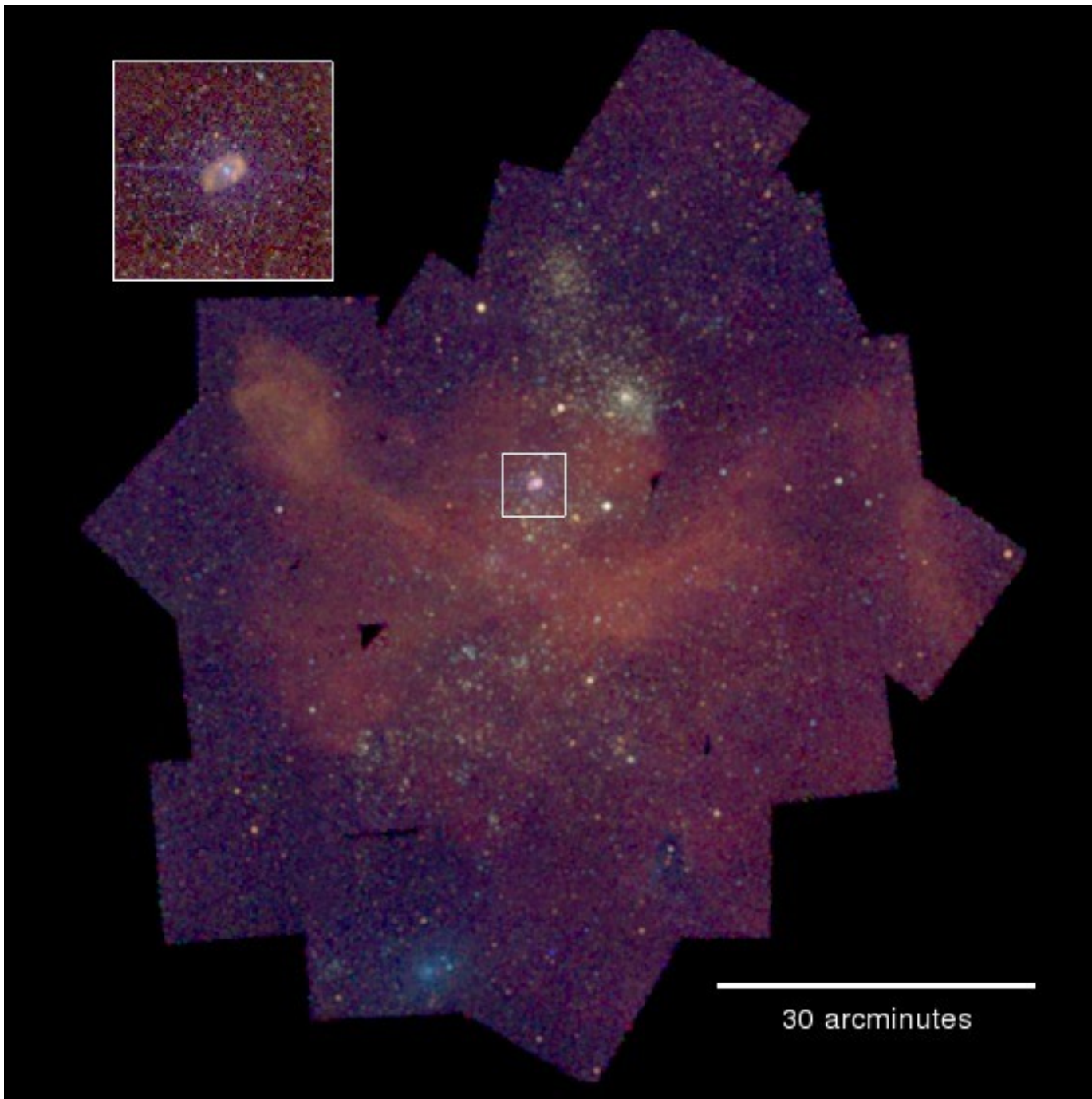
Eta Carina

Raw counts  
(left)

Exposure map  
(right)



## Combining Observations – Example 2 cont



Eta Carina

40 ACIS-I datasets 1999-2008

Mix of FAINT and VFAINT

Exposure times from 10 to 90 ks

Input was simple list of event files:

```
ls */*evt2* > lis
```

```
merge_obs @lis"[ccd_id=:3]" out
```

Result is a set of 1363 x 1537 pixel images (size autocalculated to cover the field)

# Grating data

---

## chandra\_repro:

- extracts PHA2 file
- recent mod to retain manual V&V extraction region rather than overwrite
- plan to enhance to include responses for each arm and order

## tgextract2

- extract spectra with customized source, bkg extraction regions
- useful for multiple source case

## combine\_grating\_spectra

- coming soon, will coadd spectra and weight responses for
  - multiple orders
  - multiple exposures

## tg\_findzo:

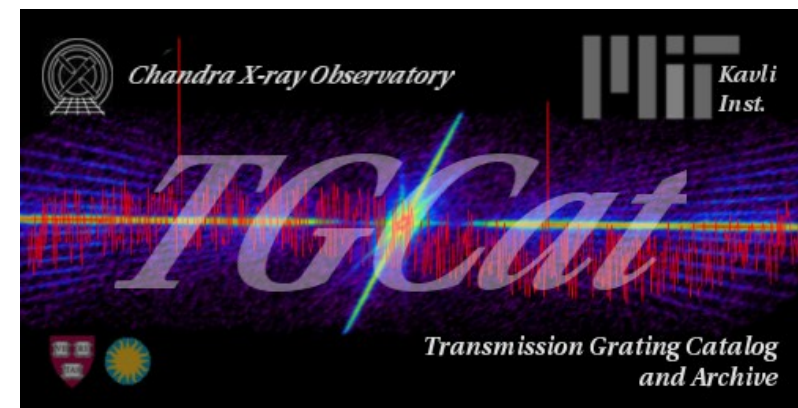
- methods to find zero order pos even when center is blanked or piled

## TGCAT (Huenemorder et al)

- [tgcatalog.mit.edu](http://tgcatalog.mit.edu)

Processed grating archive -

- manually optimized extraction regions
- extractions for almost all grating observations
- high level extracted properties



# Fluxes

---

## specextract -

- Source and background ACIS spectra for point and extended cases
  - Weighted or unweighted ARF and RMF, grouped spectra
- BUT: still sometimes awkward to use

will improve to automatically locate auxiliary files if chandra\_repro has been used

## combine\_spectra

- sum multiple imaging PHA spectra, responses  
(better to do independent fits but more convenient at low S/N)

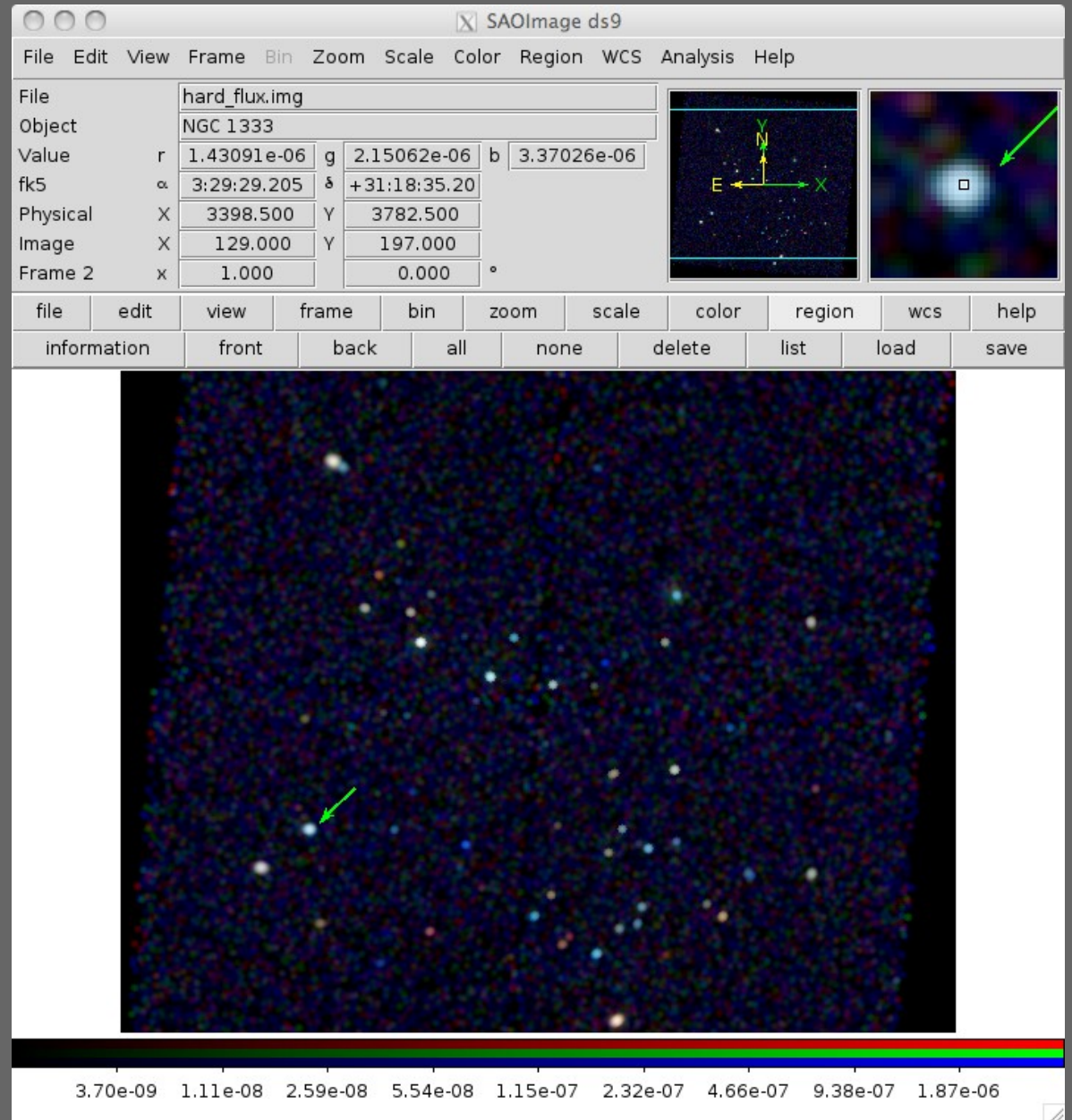
\*new\* higher level script **srcflux** which wraps use of several existing CIAO tools and scripts

```
srcflux evt2.fits ra,dec src.out
```

- generate regions using typical psf size
  - use **aprates** to determine count rates and confidence intervals (or upper limits)
  - run specextract to generate responses
  - use **eff2evt** to estimate fluxes
  - use **modelflux** to estimate fluxes given spectral model
-



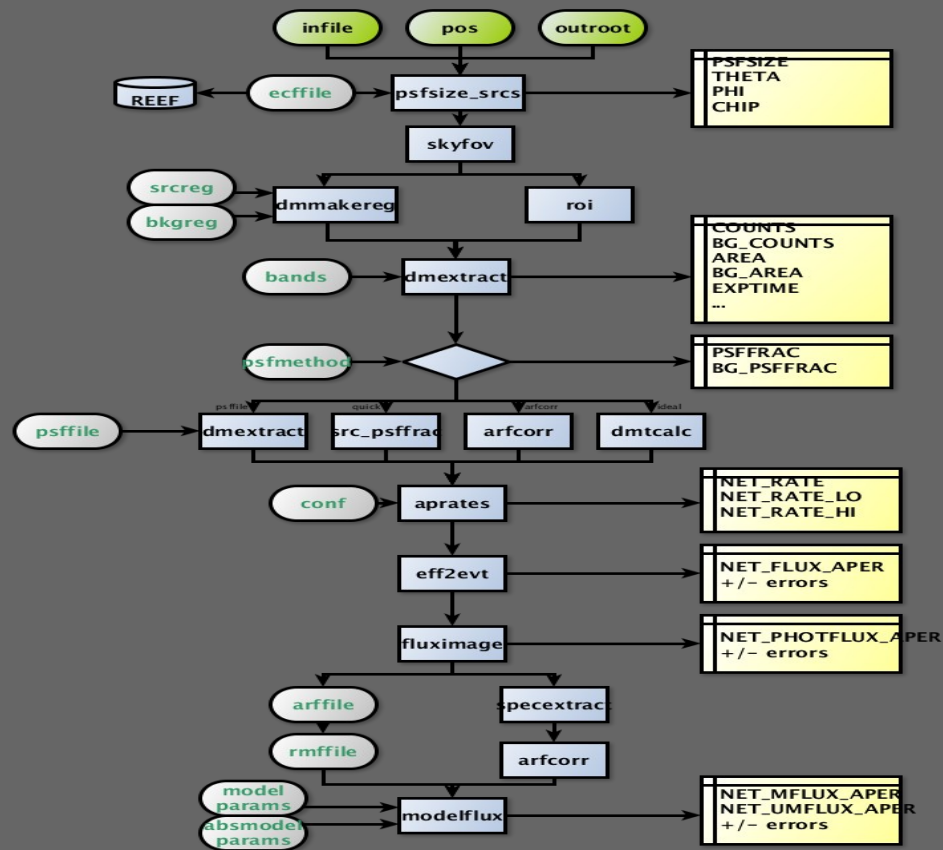
# Calculating Source Flux



What is the X-ray flux of this source?

# Calculating Source Flux

```
$ srcflux myevt2.fits "03:29:29.250 +31:18:34.73" myflux
```



Encodes the logic described in six different CIAO threads.

Return count rates and fluxes and errors with all appropriate corrections.

Uses many tools written for the Chandra Source Catalog.

Complementary to it for special cases and fields not covered by the catalog.

## Summary of source fluxes

Position	0.5 - 7.0 keV
	Value 90% Conf Interval
3 29 29.25 +31 18 34.7	Rate 0.0398 c/s (0.0381,0.0415)
	Flux 5.17E-13 erg/cm2/s (4.94E-13,5.39E-13)
	Mod.Flux 4.38E-13 erg/cm2/s (4.2E-13,4.57E-13)

```
% srcflux repro/acisf06436_repro_evt2.fits "03:29:17.653 +31:22:44.97" mysrc
srcflux
    infile = repro/acisf06436_repro_evt2.fits
      pos = 03:29:17.653 +31:22:44.97
  outroot = mysrc
    bands = broad
   srcreg =
   bkgreg =
  bkgresp = yes
psfmethod = ideal
  psffile =
    conf = 0.9
  rmffile =
  arffile =
    model = xsphabs.abs1*xspowerlaw.pow1
paramvals = abs1.nH=0.0;pow1.PhoIndex=2.0
  absmodel =
absparams =
  abund = angr
  fovfile =
  asolfile =
  pbkfile =
  mskfile =
  bpixfile =
  ecffile = CALDB
parallel = yes
  nproc = INDEF
  tmpdir = /tmp
  clobber = no
  verbose = 1
    mode = ql
```

- echoes param choices

```

Extracting counts
Setting Ideal PSF : alpha=1 , beta=0
Getting net rate and confidence limits
Getting model independent fluxes
Getting model fluxes
Getting photon fluxes
Running tasks in parallel with 4 processors.
Running eff2evt for mysrc_broad_0001_src.dat
Running aprates for mysrc_broad0001_rates.par
Running eff2evt for mysrc_broad_0001_bkg.dat
Making response files for mysrc_0001
Running modeflux for region 1
Adding net rates to output
Appending flux results onto output
Appending photflux results onto output
Computing Net fluxes
Adding model fluxes to output
Scaling model flux confidence limits

```

reports progress and results  
creates FITS output table for  
each energy band  
file has fluxes and many  
additional cols with supporting  
data

Summary of source fluxes

Position							0.5 - 7.0 keV	
							Value	90% Conf Interval
3	29	17.65	+31	22	44.9	Rate	0.0609 c/s	(0.0587,0.063)
						Flux	5.43E-13 erg/cm2/s	(5.24E-13,5.62E-13)
						Mod.Flux	5.88E-13 erg/cm2/s	(5.67E-13,6.08E-13)

~~~

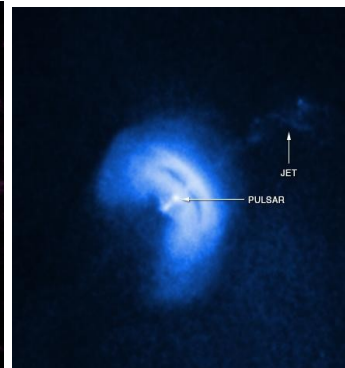
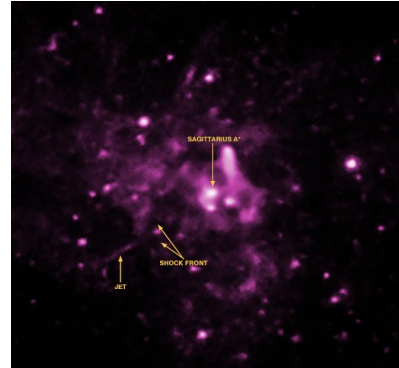
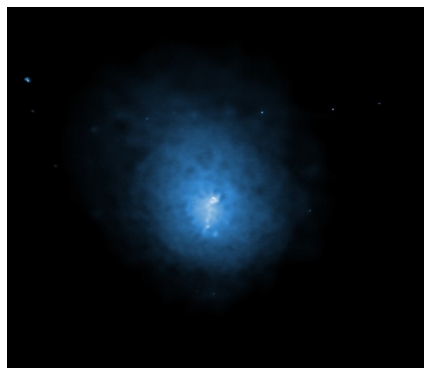
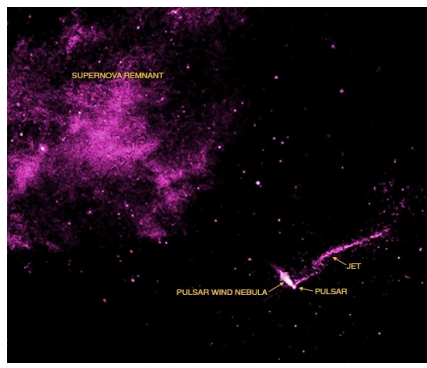
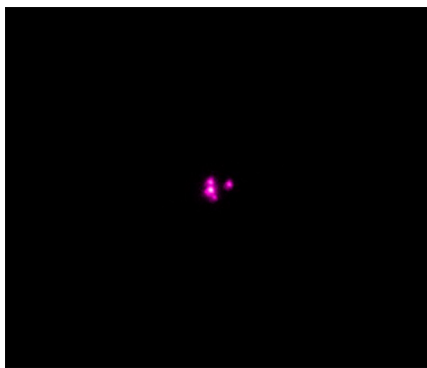
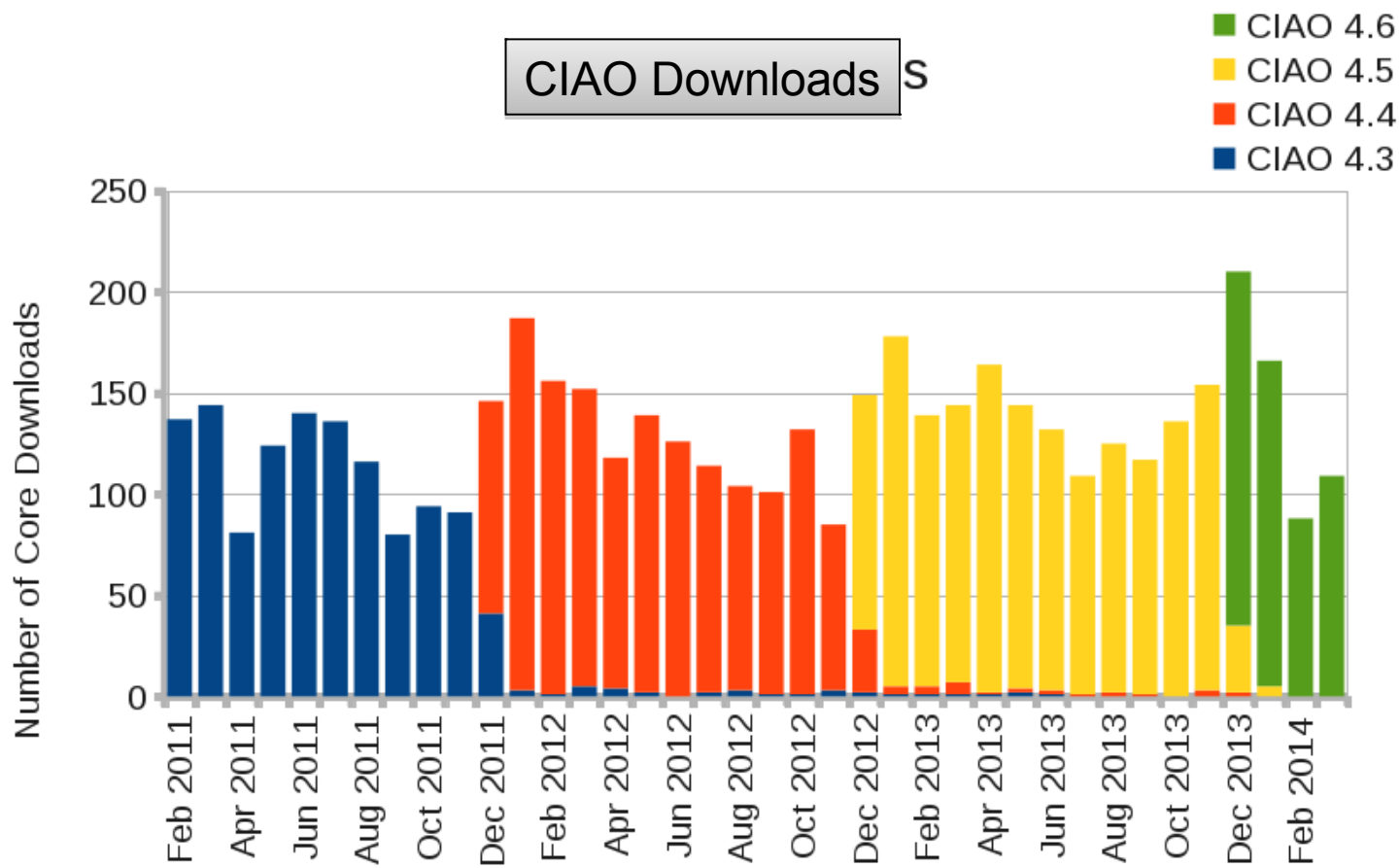


## srcflux capabilities

- finds auxiliary files automatically, like specextract
- automatically determines PSF-appropriate extraction region size for source and background, or accepts user choice
- uses one of four methods to apply aperture correction
- runs on multiple energy bands including named CSC bands
- accepts one position or a list (catalog of sources)
- calculates count rates using aprates method
- calculates fluxes two different ways (specified spectral model and eff2evt method; however, no spectral fit is performed)
- generates spectral responses for further analysis

Ongoing work: handling of warning flags for hard cases, e.g. chip edge

# CIAO continues to support Chandra science



2013-14 Press Release Images

