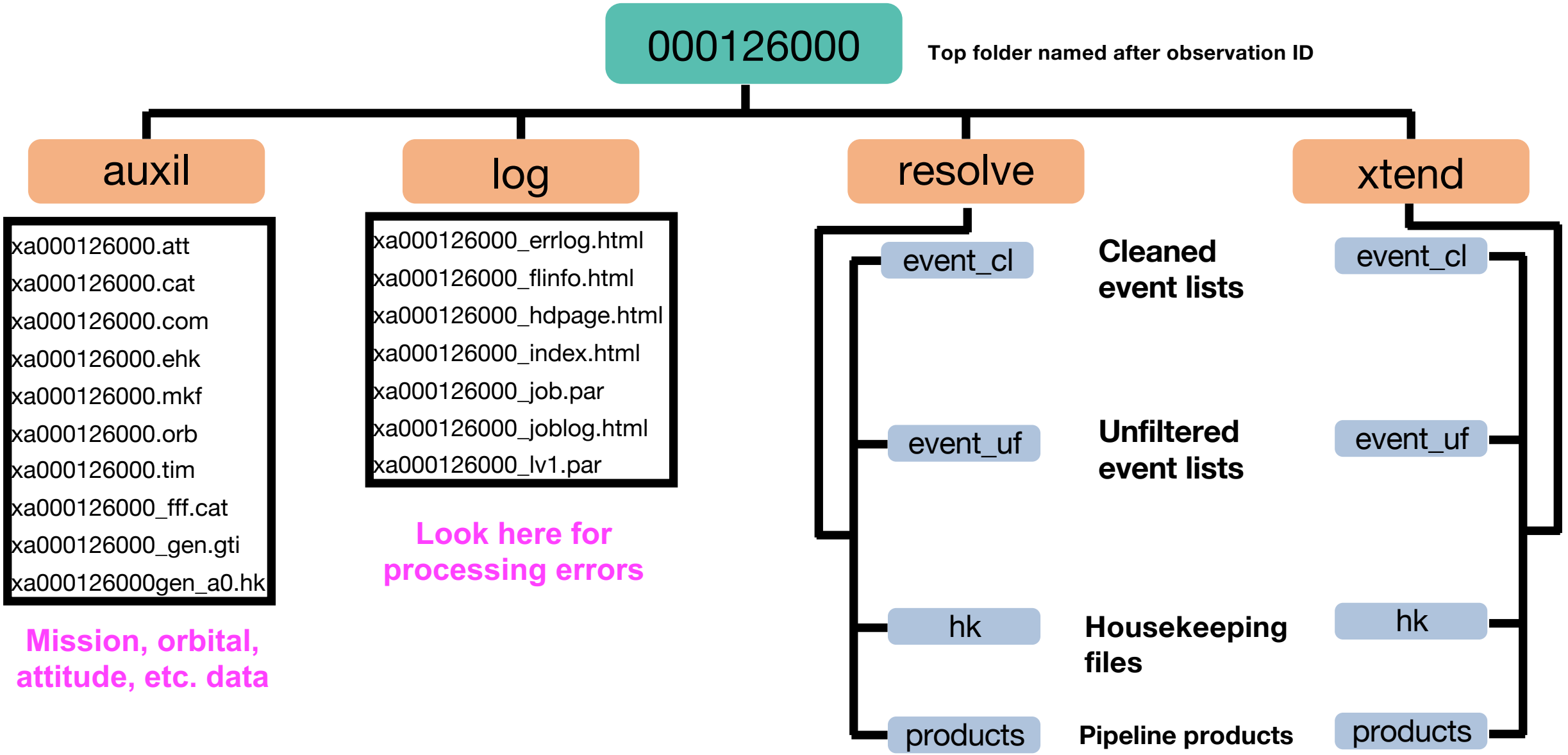
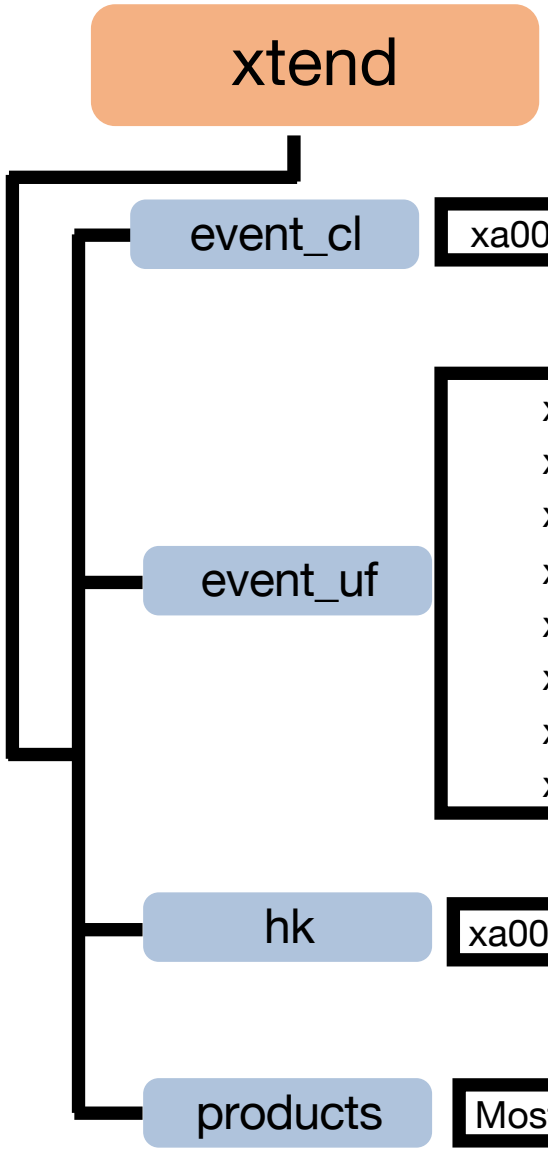
XRISM Xtend data analysis

Eric Miller, Kenji Hamaguchi
for the
XRISM SDC and GOF

Overall data directory structure (refresher)



Xtend data directory structure



'Cleaned', calibrated data — (almost) ready to analyze!

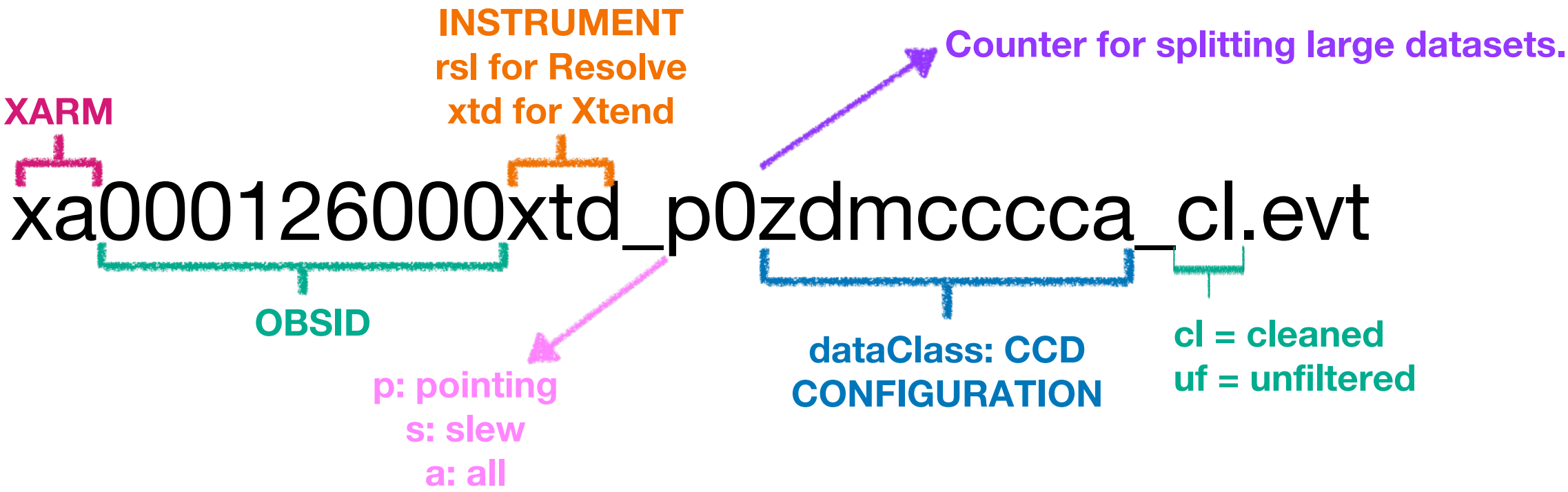
xa000126000xtd_a030000010.fpix
xa000126000xtd_a030000010.hpix
xa000126000xtd_a030100000.hpix
xa000126000xtd_a031300010.hpix
xa000126000xtd_a032000010.hpix
xa000126000xtd_a0exp.fits
xa000126000xtd_mode.gti
xa000126000xtd_p030000010.bimg
xa000126000xtd_p030000010_uf.evt
xa000126000xtd_p030100000_uf.evt
xa000126000xtd_s030000010.bimg
xa000126000xtd_s030000010_uf.evt
xa000126000xtd_s030100000_uf.evt
xa000126000xtd_seg.gti
xa000126000xtd_tel.gti

Calibrated but unfiltered data with supporting files. This is what you recalibrate and rescreen.

Instrument housekeeping, some used in reprocessing.

Quick-look image, light curves, and spectra. *Not* for analysis!

Xtend filename convention



dataClass Keys:

z = placeholder always set to 3 for Xtend in-flight data
d = DETNAM: 0 = CCD, 1 = CCD12, 2 = CCD34
m = DATAMODE: 0 = WINDOW1 (full window), 1 = WINDOW2 (1/8 window), 2 = WINDOW1BURST (full window + 0.1 sec burst mode), 3 = WINDOW2BURST (1/8 window + 0.1 sec burst mode)
cccc = hexadecimal encoding of on-board instrument settings (thresholds, area discrimination, ADC chains used, readout nodes used, charge injection settings, etc.)
a = reserved bit, 0 for now

Xtend instrument modes

- There will be a separate event file for each mode:

Mode	DATAMODE	Filename	CCDs in evt list	Frame operation
Full window	WINDOW1	*xtd_p0300* *xtd_p0320* *xtd_p0310*	CCD 1-4 (all) CCD 3,4 CCD 1,2 (rare!)	4 s exposure of full CCD
1/8 window	WINDOW2	*xtd_p0311*	CCD 1,2	0.5 s exposure of 1/8 CCD
1/8 window+burst	WINDOW2BURST	*xtd_p0313*	CCD 1,2	0.1 s exposure of 1/8 CCD with 0.4 s deadtime
Full window+burst	WINDOW1BURST	*xtd_p0312*	CCD 1,2	0.1 s exposure of full CCD with 3.9 s deadtime (‘Crab’ mode; only IT can use)
Erasing	N/A	*xtd_p0301*	N/A	Only in ‘event_uf’, do not use!

- If full window is used, there will be a single event file with all four CCDs.
  - In rare cases, CCD1,2 may be present in full window and CCD3,4 absent.
- If another mode is used, CCD1,2 will be in one event file, CCD3,4 will be in another. They must be analyzed separately.
  - CCD3,4 are ***always*** operated in full window mode.

Xtend event list contents


```
flippy> ftlist xa000126000xtd_p030000010_cl.evt h
        Name               Type       Dimensions
        ----               ----       ----------
HDU 1   Primary Array      Null Array
HDU 2   EVENTS             BinTable    40 cols x 3938313 rows
HDU 3   GTI                BinTable     2 cols x 23062 rows

flippy> ftlist xa000137000rsl_p0px1000_cl.evt.gz c
HDU 2
  Col  Name             Format[Units](Range)      Comment
    1 TIME              1D [s]                Seconds from 01 Jan 2019 00:00:00
   12 CCD_ID            1B (0:3)              '0: CCD1, 1: CCD2, 2: CCD3, 3: CCD4'
   13 CCD_NAME          8A                    'CCD Name'
   14 SEGMENT           1B (0:1)              '0: AB, 1: CD (Segment ID)'
   17 READNODE          1B (0:1)              '0: A or D, 1: B or C (Readout Node)'
   27 DETX              1I (1:1810)           'Pixel X on DET-Coordinate'
   28 DETY              1I (1:1810)           'Pixel Y on DET-Coordinate'
   31 X                 1I (1:2430)           'Pixel X on SKY-Coordinate'
   32 Y                 1I (1:2430)           'Pixel Y on SKY-Coordinate'
   36 PHA               1I (0:4095)           'Pulse Height Amplitude Sum of 3x3 Pixels'
   37 PI                1I (0:4095)           'Pulse Height Invariant'
   38 GRADE             1I                    'Grade Value for Pixel Hit Pattern'
   39 STATUS            48X                   'Event Flag'
   40 PROC_STATUS       32X                   'Record Bad Telemetry or Bad Values'
HDU 3
  Col  Name             Format[Units](Range)      Comment
    1 START             D [s]                 GTI start time
    2 STOP              D [s]                 GTI stop  time
```

Each row is an event, ideally from a single X-ray.

You can filter on

- time
- energy
- region
- quality

to create

- images
- light curves
- spectra

Xtend event list in detail

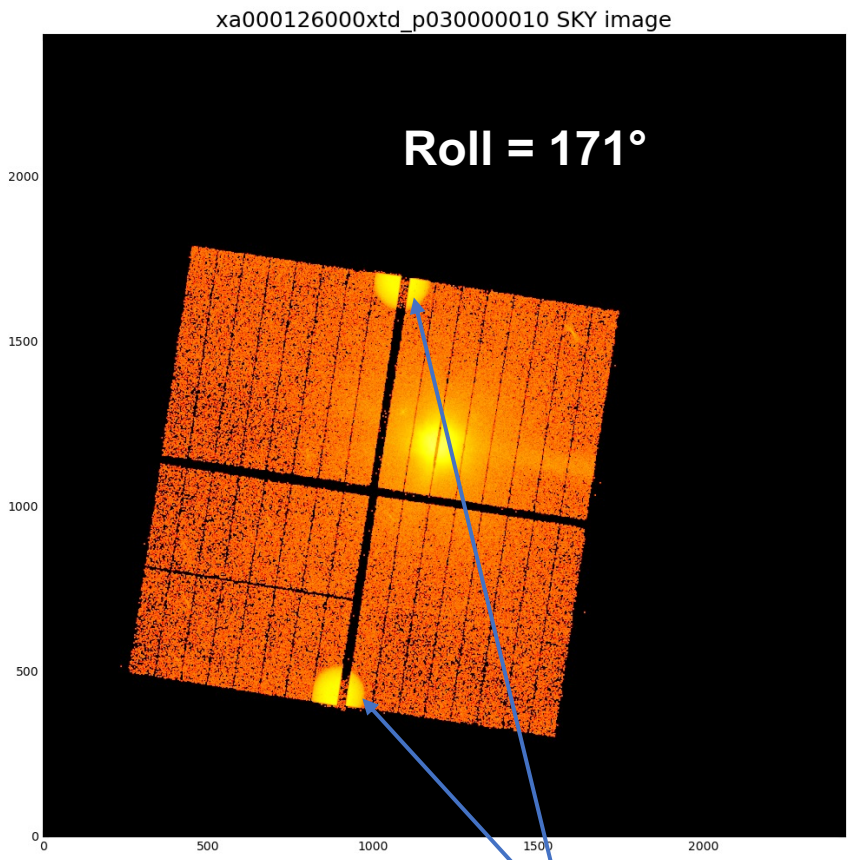
Xtend Event File
Columns of Interest

Column	Description	Range
CCID_ID	ID number of the CCD on which the event is found Aim-point CCD is CCD_ID = 1	0-3
CCD_NAME	Name of the CCD on which the event is found; different numbering than CCD_ID!!! Aim-point CCD is CCD_NAME = CCD2	CCD1-4
SEGMENT	One of two segments on each CCD	0-1
READNODE	One of two possible read-out nodes for each segment. Only one is used at a time and is expected not to change.	0-1
RAWX,Y	Coordinates relative to readout node.	0-319,0-639
ACTX,Y	Coordinates relative to the physical CCD (look-down).	1-640,1-640
DETX,Y	Coordinates relative to the four CCDs (look-up).	1-1810
FOCX,Y	DET coordinates shifted to a common aimpoint with Resolve (look-up).	1-2430
X,Y	Coordinates relative to the spacecraft attitude; used with WCS to display RA,DEC	1-2430
GRADE	Pattern of 5x5 pixel island with signal. From ASCA, Suzaku, Hitomi. Good grades are 0,2,3,4,6 (singles, doubles, triples, quads)	0-11
PI	Linearized Energy Channel	0-6000
STATUS	48 bit Event Flag — discussed later	0-1

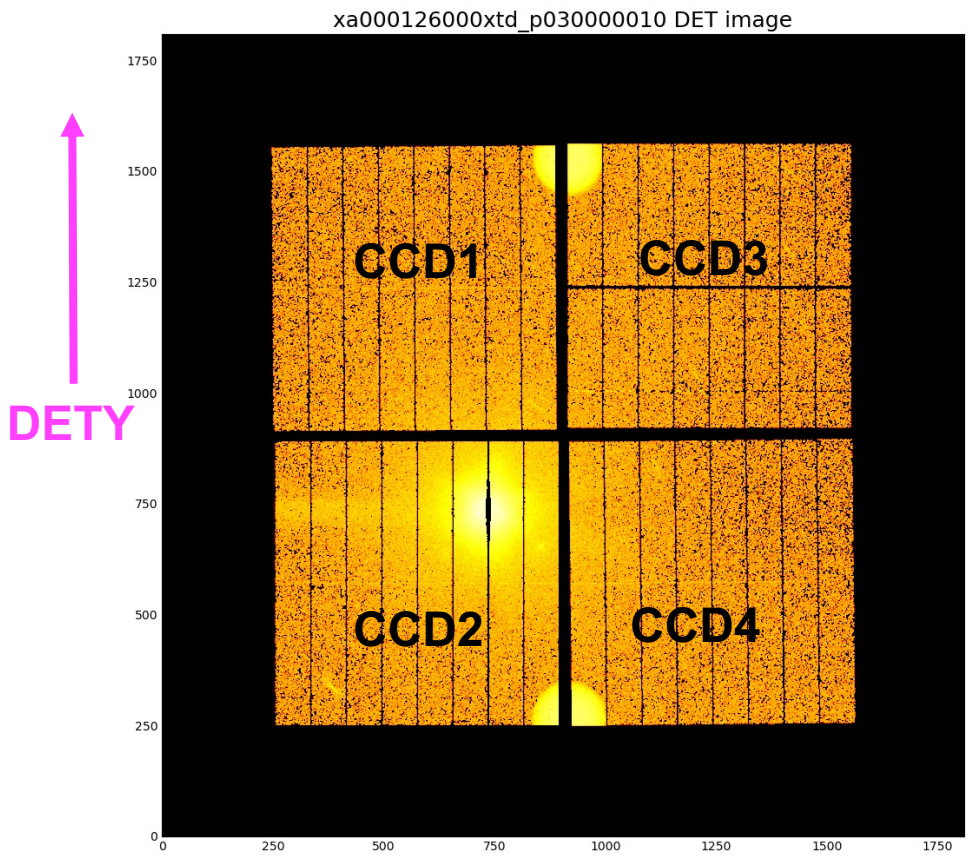
Xtend “look-up” coordinates

X-Ray Imaging and Spectroscopy Mission

X,Y (SKY)



DET



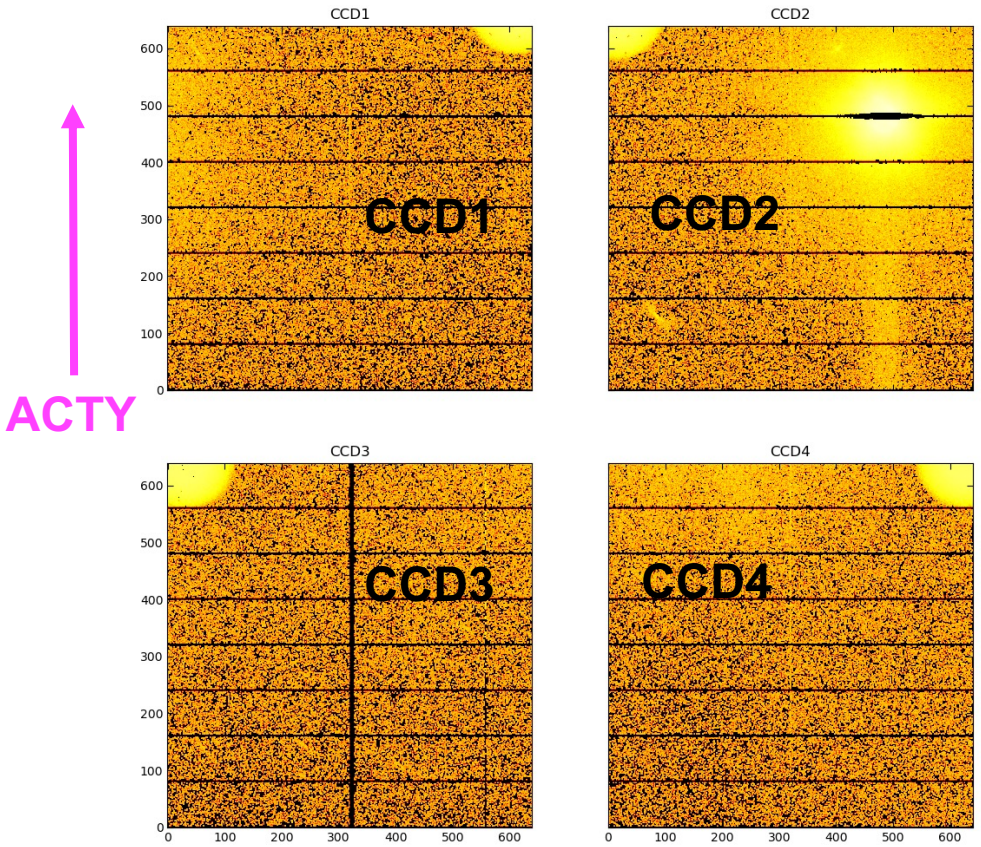
^{55}Fe calibration sources produce lines of Mn Kα (5.9 keV) and Kβ (6.4 keV) plus bremsstrahlung continuum — recommend you mask these regions!

Xtend “look-down” coordinates

XRISM X-Ray Imaging and Spectroscopy Mission

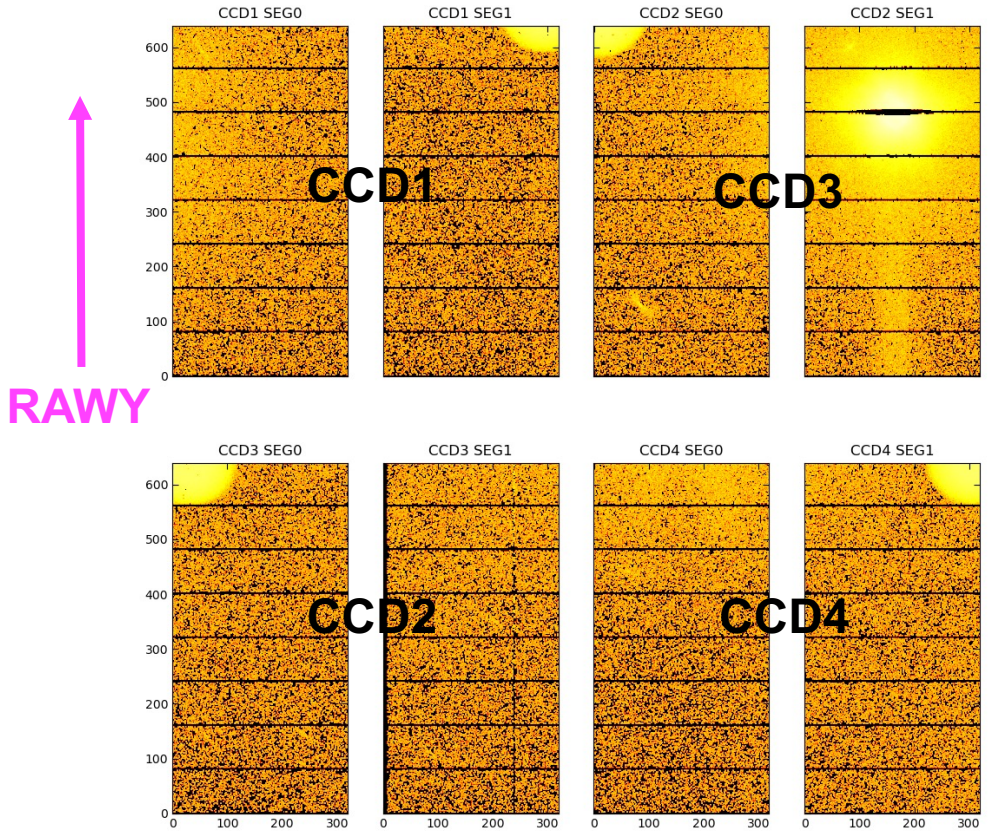
ACT

xa000126000xtd_p030000010 ACT images



RAW

xa000126000xtd_p030000010 RAW images



Xtend pipeline processing tasks

- coordevt
  - Converts RAW coordinates to ACT, DET, FOC, SKY.
- xtdpi
  - Corrects pixel ‘pulse height amplitudes’ or PHAS for column-to-column gain differences, charge trailing, charge-transfer inefficiency (CTI).
  - Assigns a GRADE to each event based on the pattern of 5x5 pixels above threshold.
  - Sums pixel PHAS to calculate event PHA, depending on GRADE rules.
  - Correct PHA for overall gain to produce ’pulse invariant’ PI value, in bins of 6 eV.
  - Runs TWICE to apply GRADE-dependent corrections and PI-dependent split threshold.
- xtdflagpix
  - Set data quality STATUS flags for future screening to create cleaned events.
  - Runs TWICE to set flags for flickering pixels, but that will change (more details later).

…plus a few other things. See `fhelp xtdpipeline` for full details.

Xtend pipeline screening

- `ahgtigen` creates Good Time Intervals (GTIs) based on some elements of instrument or spacecraft housekeeping (HK)
- `ahscreen` applies those GTIs and other event-based filtering to produce the cleaned events

GTI	Nominal instrument status — from HK (MKF)
GTI	Telemetry is not saturated
GTI	Pointing is accurate, stable — from attitude, angular distance (EHK)
GTI	Safe angle above Earth and sunlit Earth limb — from orbit (EHK)
GTI	Away from South Atlantic Anomaly — from orbit (EHK)
EVENT	GRADE == 0,2,3,4,6
EVENT	Data quality based on STATUS (see next)

Xtend event STATUS (1/2)

```
==========================================================================
1       Any of the flags specified in 'bad_status' are set; a bad event
2       Inside the calibration source region
--------------------------------------------------------------------------
3       Out of CCD                                          (Out of area)
4       Out of window
5       Out of area discrimination
--------------------------------------------------------------------------
6       CI row                                                 (Pixels)
7       Bad pixel from CalDB
8       Bad column from CalDB
9       Hot pixel from pre-pipeline
10      Flickering pixel
--------------------------------------------------------------------------
11      CCD boundary                                        (Boundaries)
12      Window boundary
13      Segment boundary
14      Area discrimination boundary
15      At least one 3x3 surrounding pixel has a bad status
--------------------------------------------------------------------------
16      CI trailing row in rows specified by 'citrailnbr'      (Neighbors)
17      CI preceding row in rows specified by 'ciprenbr'
18      Preceding/following pixel in a partial bad column
19      Neighbors of bad/hot pixel and bad column
20      Neighbors of flickering pixel
21      Neighbors preceding/following a bad column
22      Neighbors of CCD/window boundary
23      Neighbors of segment boundary
```

From ‘fhelp xtdflagpix’

Events with these STATUS flags are considered “bad” by default.

Xtend event STATUS (2/2)

24	(xtdphas) 3x3 info is present, but 5x5 is absent	(Others)
25	(xtdphas) 3x3 is absent	
26	(xtdpi – general) PHAS[0] < event threshold	
27	(xtdpi – vtevnodd) Video temperature is out of range	
28	(xtdpi – vtevnodd) Lack of video temp HK at time close to the event	
29	(xtdpi – chtrail/CTI) Correction value is negative	
30	(xtdpi – general) Null value by correction process	
31	1st trailing row of the CI rows	(Diagnostics)
32	1st preceding row of the CI rows	
33	2nd trailing row of the CI rows	
34	2nd preceding row of the CI rows	
35	3rd trailing row of the CI rows	
36	3rd preceding row of the CI rows	
37	Cosmic ray echo pixel	(Cosmic Rays)
38–48	Reserved	

Events with these STATUS flags are considered “bad” by default.

A note about changing ‘bad’ events

- Data can be rescreened to exclude events with different STATUS.
  - Remove calibration source regions (better to do this with a region).
  - Remove more neighboring rows of charge injection rows.
  - In general, this shouldn’t be necessary.
- It is not sufficient to simply re-screen the cleaned or unfiltered event list.
  - `xtdflagpix` creates a ‘bad pixel image’ or `.bimg` file which is used to generate the exposure map. This needs to be updated as well.
- Recommended method for re-screening — run the whole pipeline!
  - Run `xapipeline` with desired `bad_status`:

```
xapipeline indir=000126000 outdir=000126000_reproc
  steminputs=xa000126000 stemoutputs=DEFAULT
  entry_stage=1 exit_stage=2 instrument=XTEND verify_input=no
  bad_status=<whatever you want>
```

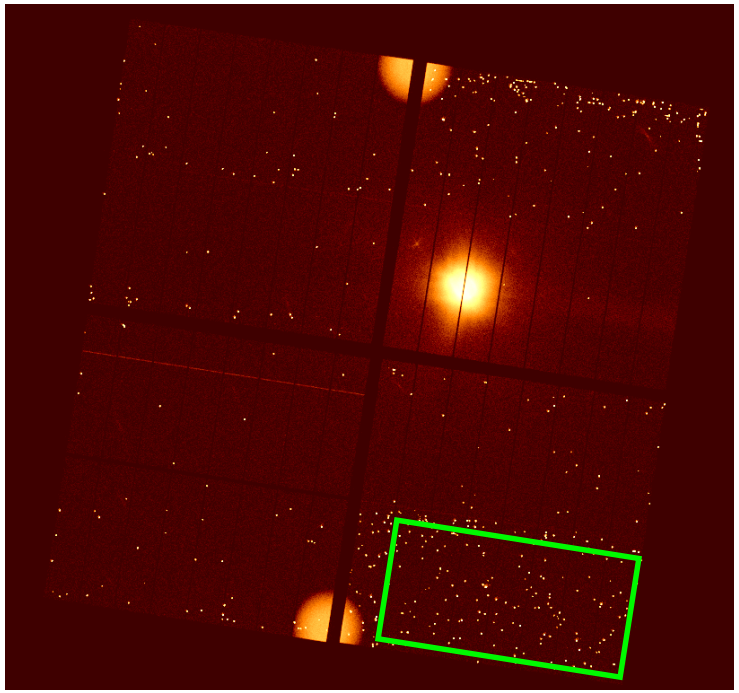
Runs `xtdflagpix`

Screens data

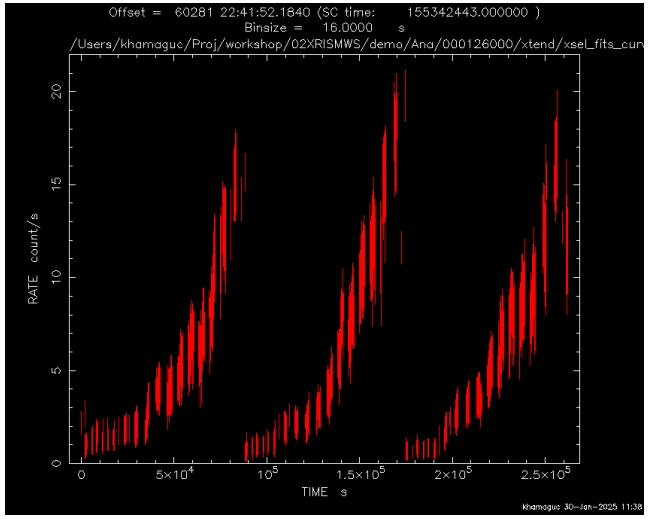
  - Default is ‘`bad_status=3:9,11:12,16:19,25:28,30,37`’
    Decode this by looking at the help for `xtdflagpix` or the previous slides

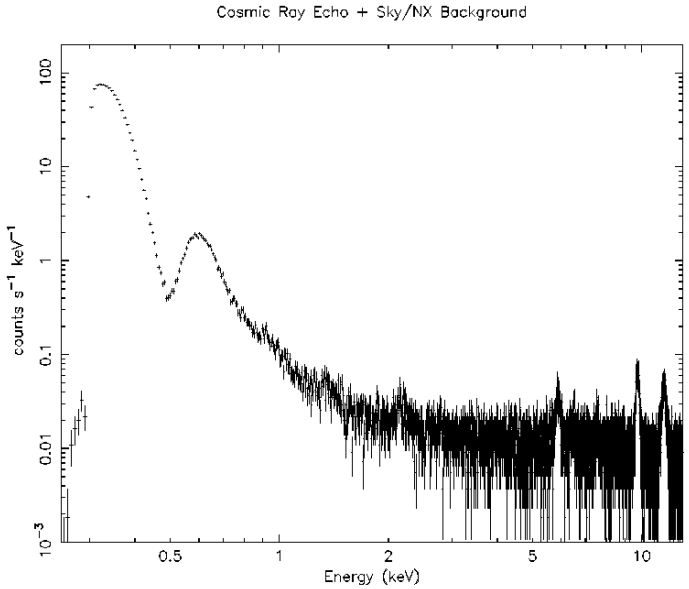
Cosmic Ray echo events (reminder)

- Few cosmic ray echo pixels near the aim point
- Those events are seen mostly below ~0.4 keV



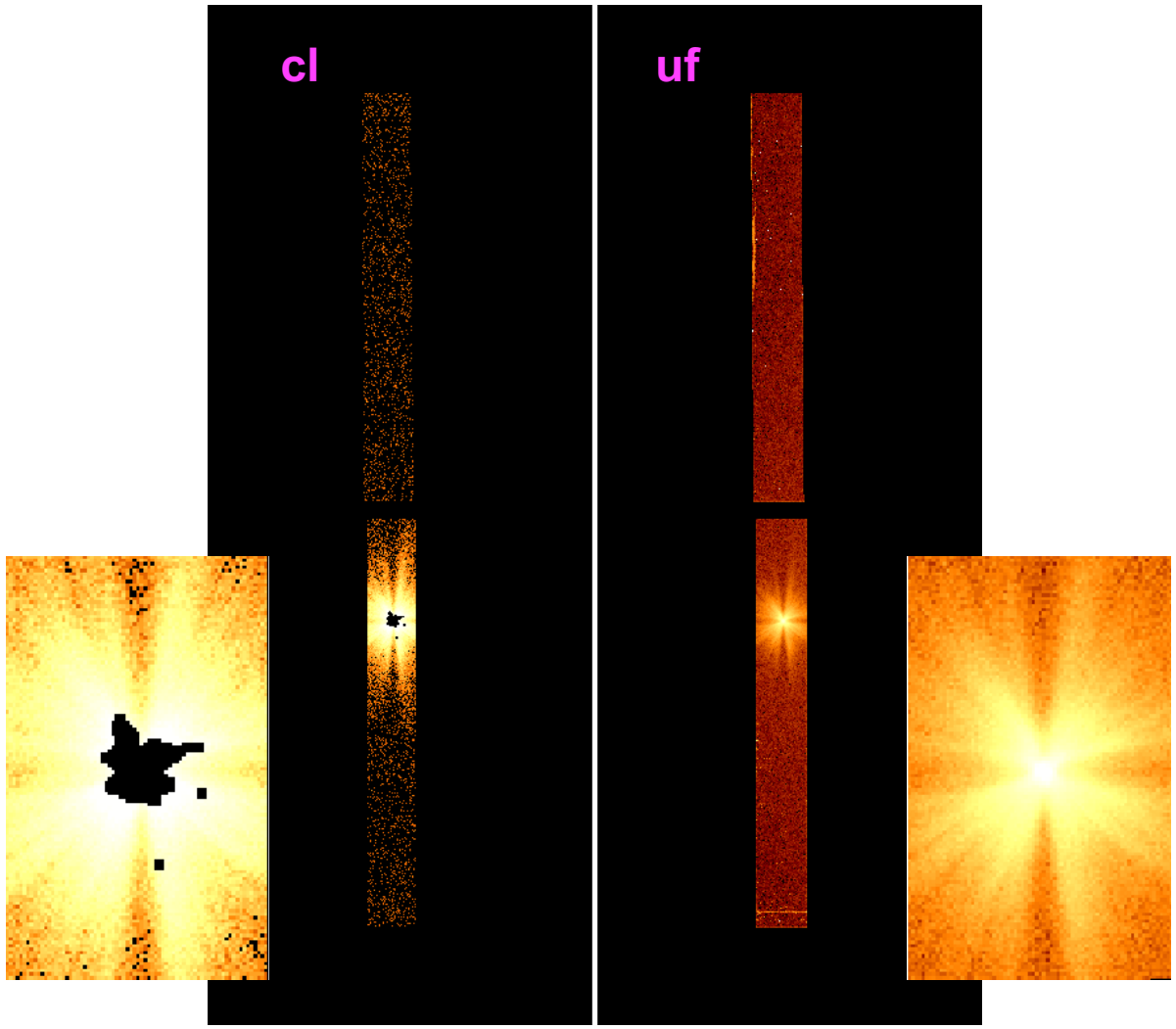
N132D (ObsID: 000126000)
no energy filter





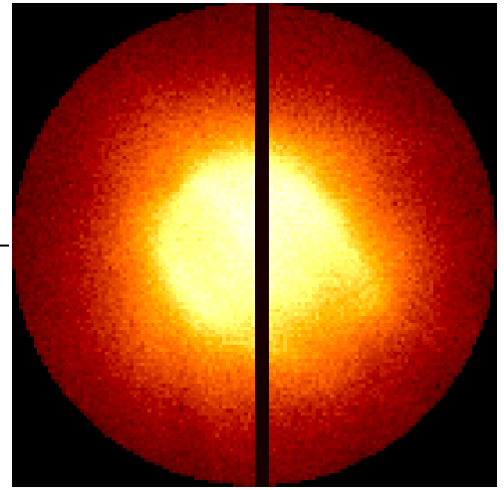
Cosmic Ray echo events

- CR echo produces persistent events in random pixels
- ‘searchflickpix’ thinks these are flickering pixels, but it runs amok
  - The core of the PSF for bright point source (and some extended sources) is eaten away in the cleaned data, resembling pile-up.
  - ‘searchflickpix’ uses an algorithm based on cleansis, with additional cleaning algorithms and enhancements.
  - There is no single set of parameters that can work for all cases (point source vs. extended source, bright vs. faint).
  - ‘searchflickpix’ was never intended to be used in this way. It will not be run in xtdpipeline after the next software release.
- ‘searchflickpix’ is still run in the pipeline, but they are ***not*** flagged or removed
- ***DO NOT USE*** the provided flickering pixel file (xa000126000xtd_a030000010.fpix)
- ‘searchflickpix’ can be used to remove them in some cases, see the Quick Start Guide
- New tool ‘xtdpixclip’ in next HEASOFT release
- Future tool to correct for them in the pipeline



Extract spectrum in Xselect

- Region can be in RA,DEC (X,Y) or DET coordinates, set it properly
- WMAP must use DET (default); this is used to weight the RMF
- Kenji will walk through this in his demo



```
 Notes: XSELECT set up for       XRISM
 Time keyword is TIME        in units of s
 Default timing binsize =    16.000

Setting...
 Image  keywords    = X          Y           with binning =    1
 WMAP   keywords    = DETX       DETY        with binning =    1
 Energy keyword    = PI                     with binning =    1

Getting Min and Max for Energy Column...
Got min and max for PI:      0    4095
```

Create the RMF

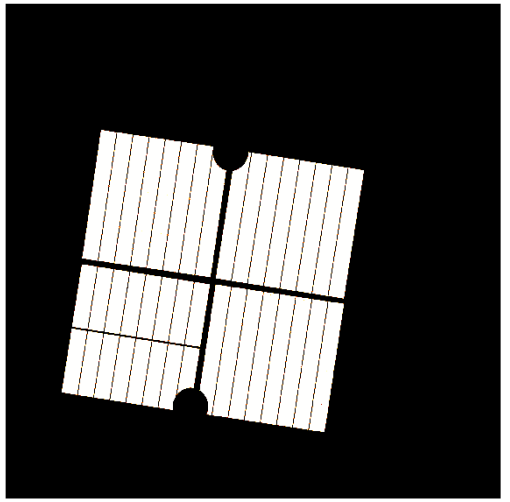
- Input to `xtdrmf` is the extracted spectrum
  - Contains the WMAP = weighted map, which is used to weight and combine responses from small regions (similar to Resolve per-pixel response weighting)
  - Also reads Xtend configuration from the header, since response depends on the window mode
- None of the other parameters need to be altered
- This runs quickly compared to rslmkrmf

```
xtdrmf \
     infile=n132d_r2m_src.pi \
     outfile=n132d_r2m_src.rmf \
     rmfparam=CALDB \
     eminin=200. dein="2,24" nchanin="5900,500" \
     eminout=0 deout=6 nchanout=4096 \
     clobber=yes mode=hl
```

Create the exposure map

- The exposure map is required input to the ARF generator.
- The event list is used as `gtifile`; these times are applied to the attitude information in `ehkfile` to “wobble” the Xtend FoV around on the sky and build up a map of how much time each pixel looked at a particular point in the sky.
- Bad pixels, chip gaps, etc. are include via `badimgfile`.
- Exposure map is agnostic to the extraction region! Can use for multiples ARFs.

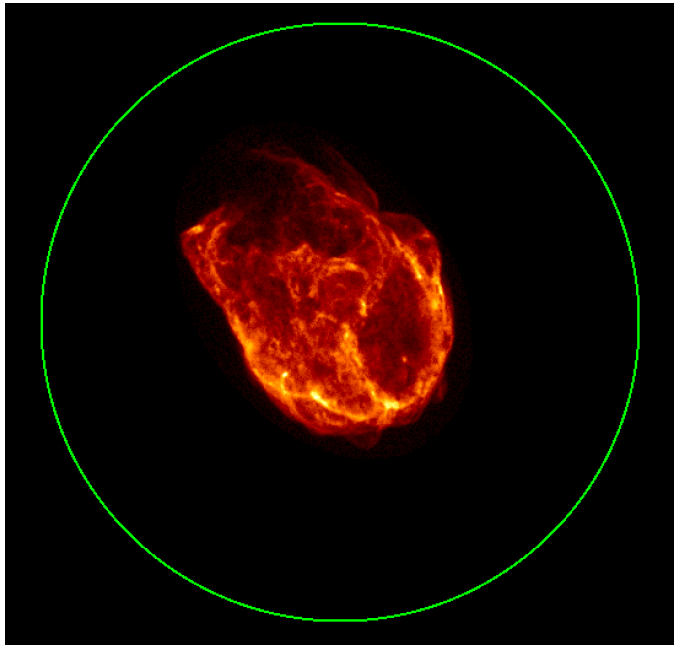
```
xaexpmap \
     instrume=XTEND \
     outfile="n132d_r2m_src.expo" \
     ehkfile="xa000126000.ehk" \
     gtifile="xa000126000xtd_p030000010_cl.evt" \
     badimgfile="xa000126000xtd_p030000010.bimg" \
     pixgtifile=NONE \
     outmaptype=EXPOSURE delta=20.0 numphi=1 stopsys=SKY \
     clobber=yes mode=hl
```



Create the ARF

- Region must be in RA,DEC or DET (*not* X,Y)
- `source_ra` and `source_dec` are not used if `sourcetype=IMAGE`, coords are read from the image (must have WCS header info)
- Here I use a small cutout of a Chandra image. Run took 16 minutes!

```
xaarfgen \
     xrtevtfile="raytrace_n132d_src.fits" \
     outfile="n132d_r2m_src.arf" \
     source_ra=0. source_dec=0. \
     telescop=XRISM instrume=XTEND \
     emapfile="n132d_r2m_src.expo" \
     rmffile="n132d_r2m_src.rmf" \
     sourcetype=IMAGE imgfile="acis_region_img.fits" \
     regmode=RADEC regionfile="n132d_r2m_src_wcs.reg" \
     erange=".3 18. .35 8." \
     numphoton=300000 minphoton=100 seed=7 \
     clobber=yes mode=hl
```



Extract an NXB spectrum

- Xtend NXB is spatially dependent — higher away from CCD readouts (high ACTY)
- regfile1 is the region to extract the NXB, regfile2 is the source extraction region
- xtdnxbgen will scale the NXB spectrum by the difference, but region coords must be same

```
xtdnxbgen \
     infile="xa000126000xtd_p030000010_cl.evt" ehkfile="xa000126000.ehk" \
     regmode=DET regfile="xtd_bgboxes_det.reg" regfile2="n132d_r2m_src_det.reg" \
     innxbfile="../../../nxb/merged_rev2_nte_xtend_fullwin_gtifix.evt" \
     innxbehk="../../../nxb/merged_reduced_fix.ehk" \
     apply_xtdtools=no \
     database=LOCAL db_location=./ \
     timefirst=-1000 timelast=+1000 \
     SORTCOL=CORTIME sortbin="0,4,6,8,10,12,99" \
     expr=TOO LONG TO SHOW \
     outpifile="n132d_r2m_src_nxb.pha" \
     outnxbfile=NONE outnxbehk=NONE \
     clobber=yes mode=hl
```