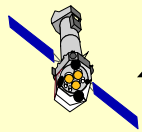
Epic Detector Matrices

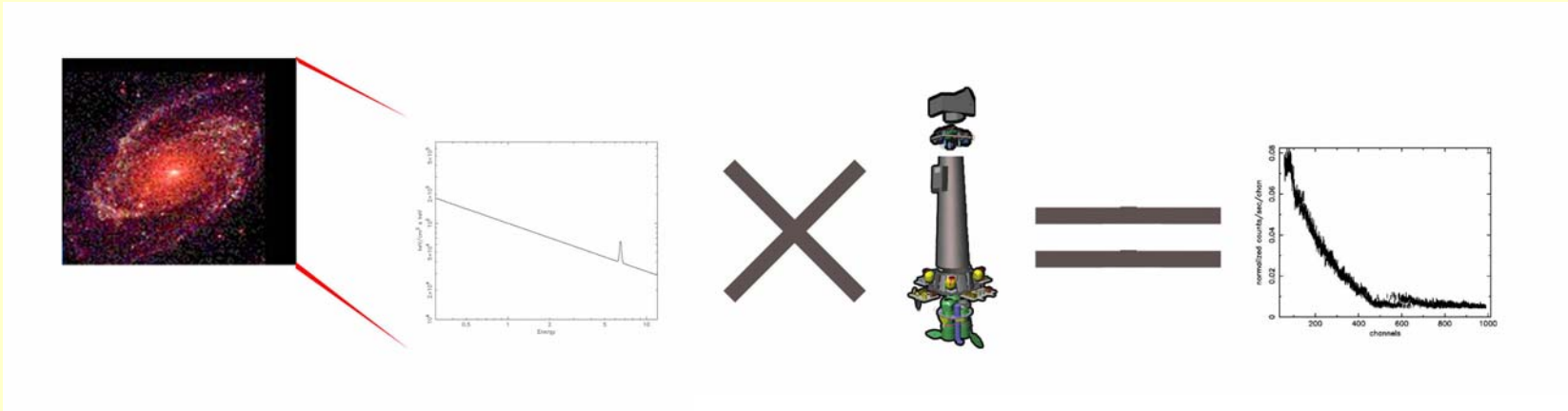
Richard Saxton

June 2008



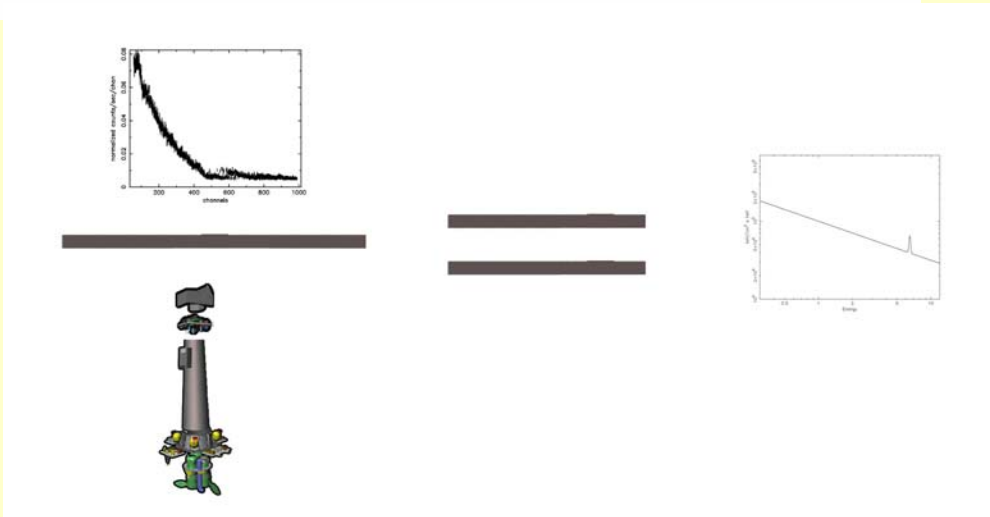
XMM-Newton

Why do we need detector matrices ?



Initial source spectrum

Source flux



What does the SAS provide ?

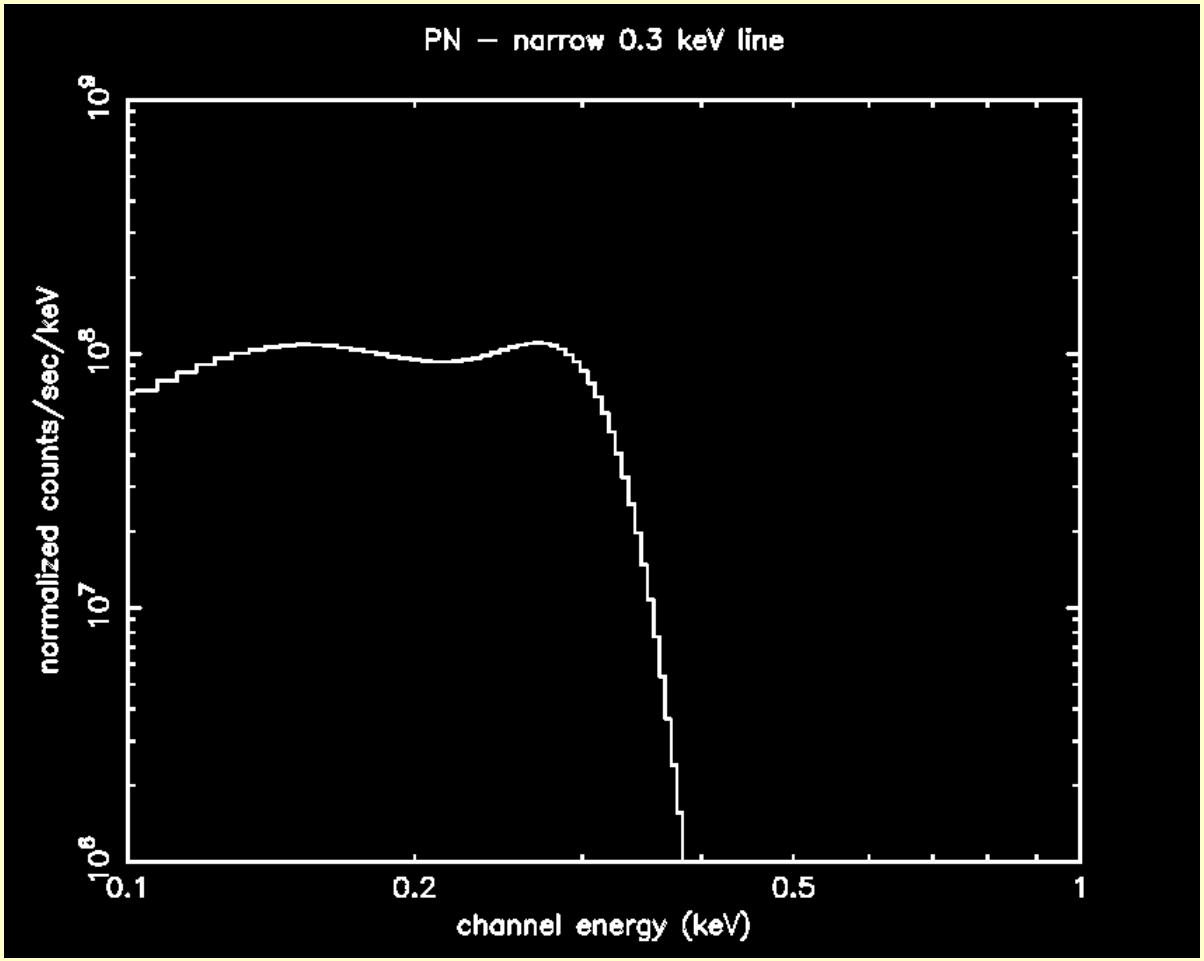
rmfgen *- Calculates the redistribution matrix (RMF)*

e.g. **rmfgen spectrumset=spectrum.ds rmfset=myspec.rmf**

arfgen *– Calculates the instrument effective area (ARF)*

e.g. **arfgen spectrumset=spectrum.ds arfset=myspec.arf**

The EPIC redistribution function



Standard Matrices

The SAS provides a set of standard RMFs to allow calibration developments to proceed independently of SAS releases.

These are available from:

http://xmm.esac.esa.int/external/xmm_sw_cal/calib/epic_files.shtml

EPIC-MOS: RMFs

Standard Matrices:

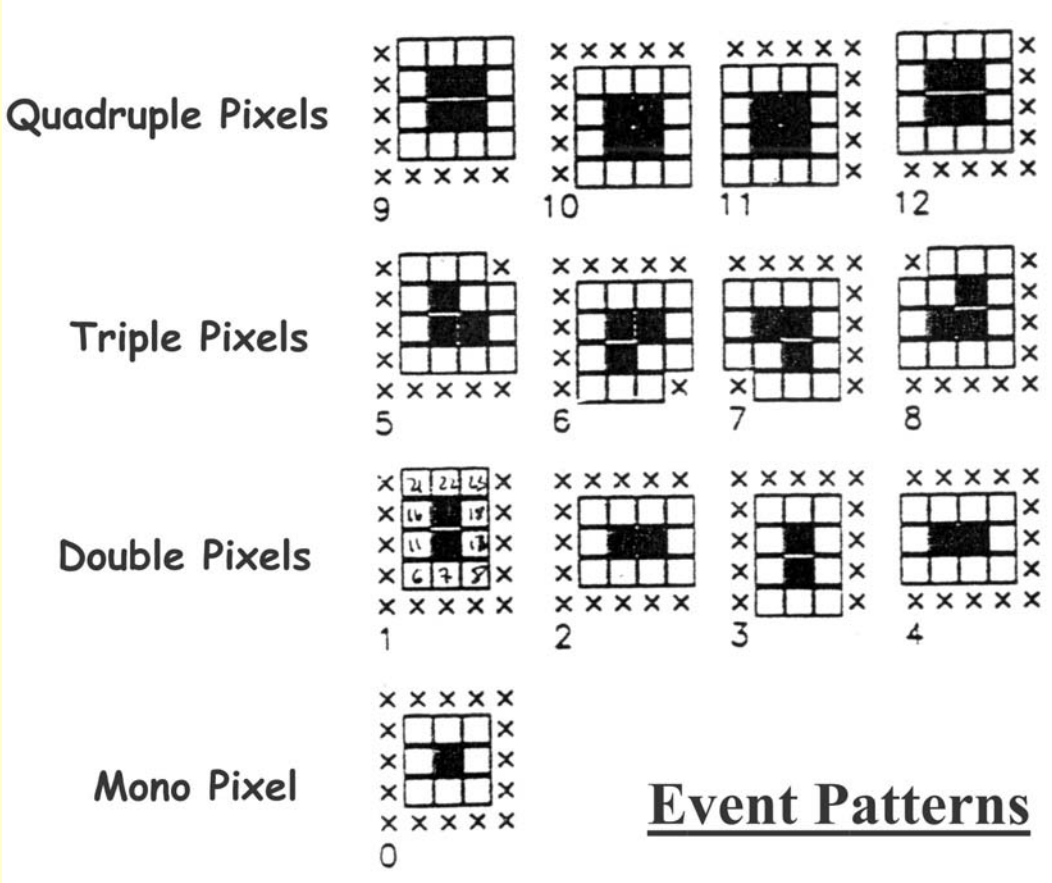
m11_110_im_pall_v1.2.rmf.gz - Mos-1, event patterns 0-12
m21_110_im_pall_v1.2.rmf.gz - Mos-2, event patterns 0-12
m11_255_im_p0_v1.2.rmf.gz - Mos-1, event pattern 0 only
m21_330_im_p0_v1.2.rmf.gz - Mos-2, event pattern 0 only
m11_534_tu_p0_v1.2.rmf.gz - Mos-1, timing mode
m21_534_tu_p0_v1.2.rmf.gz - Mos-2, timing mode

Time-dependent matrices, currently 9 epochs.

SAS command:

rmfgen spectrumset=spectrum.ds rmfset=myresp.rmf

Event Patterns



EPIC-PN: RMFs

Standard Matrices:

- ***epn_ff20_sY9.rmf*** - Full frame mode, pattern 0 only, on-axis
- ***epn_ff20_sY0.rmf*** - Full frame, pattern 0 only, at edge of field
- ***epn_ff20_dY5.rmf*** - Full frame, patterns 1-4, centre of CCD
- ***epn_ff20_sdY9.rmf*** - Full frame, patterns 0-4, on-axis
- ***epn_ef20_sY9.rmf*** - Extended full frame, pattern 0, on-axis
- ***epn_sw20_sY9.rmf*** - Small window mode, pattern 0, on-axis
- ***epn_lw20_sY9.rmf*** - Large window mode, pattern 0, on-axis
- ***epn_ti20_sY9.rmf*** - Timing mode, pattern 0, on-axis
- ***epn_bu20_sY9.rmf*** - Burst mode, pattern 0, on-axis

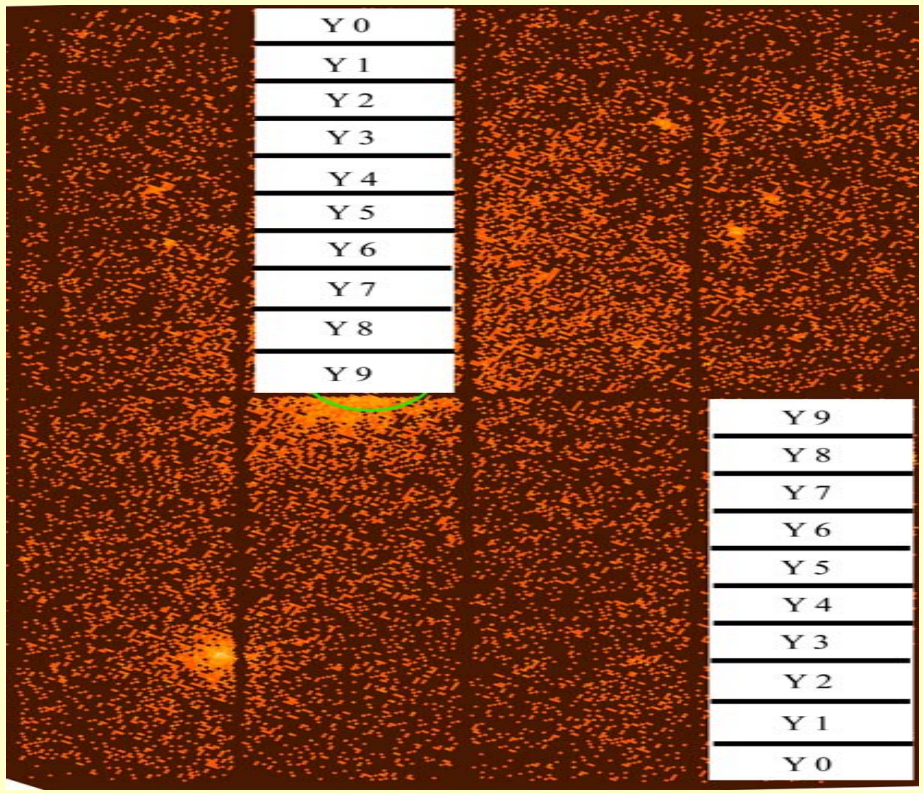
etc.

SAS command:

rmfgen spectrumset=spectrum.ds rmfset=myresp.rmf

EPIC-PN: RMFs (II)

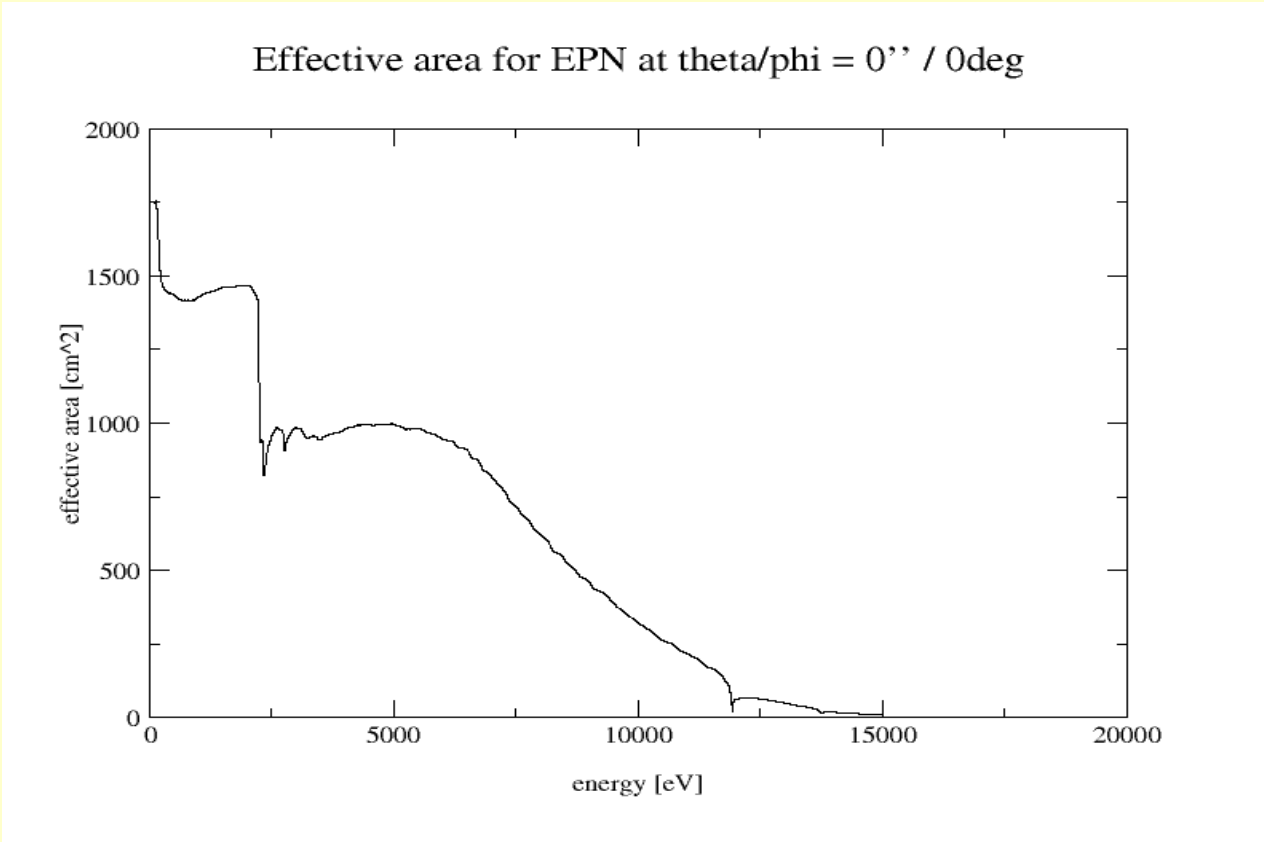
Function of observing mode, patterns, position



arfgen: effective area contributions

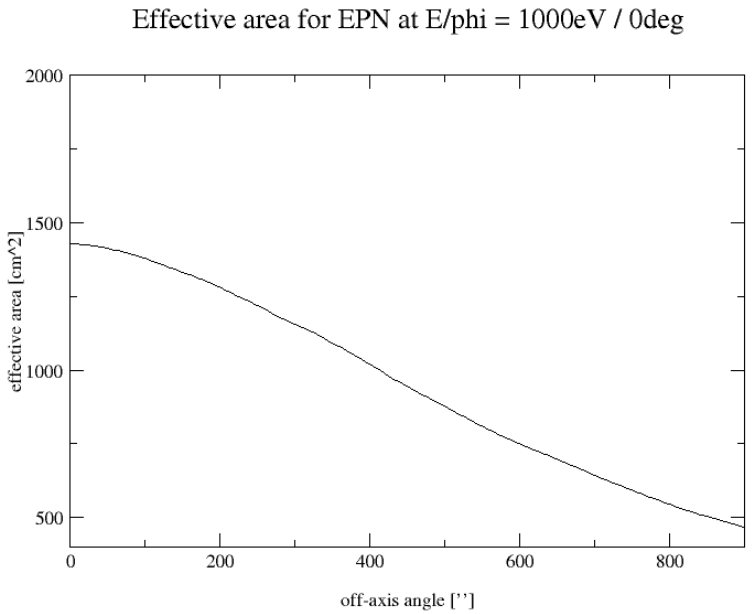
- Telescope effective area
- Vignetting
- Filter transmission
- Detector quantum efficiency
- Encircled energy correction
- Flux loss due to CCD gap, bad pixels and offset columns

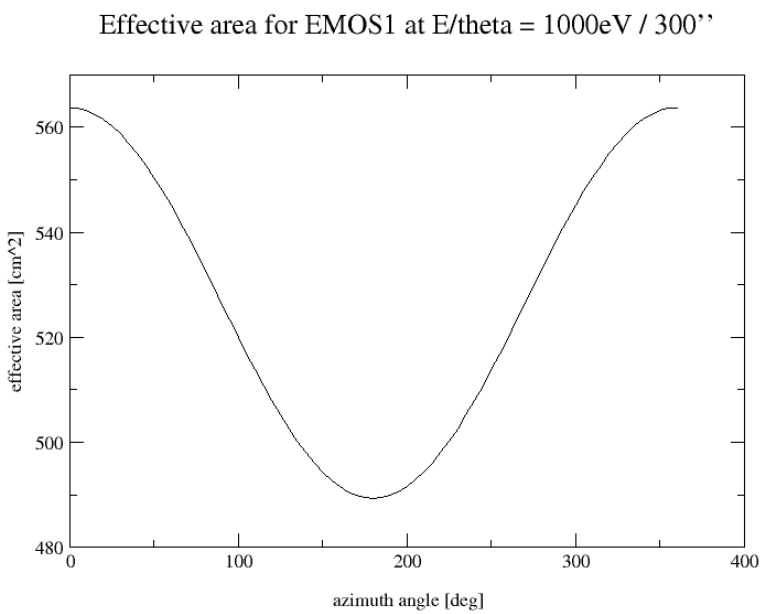
Mirror Effective Area



Option: *modeleffarea*=yes (default)

Vignetting

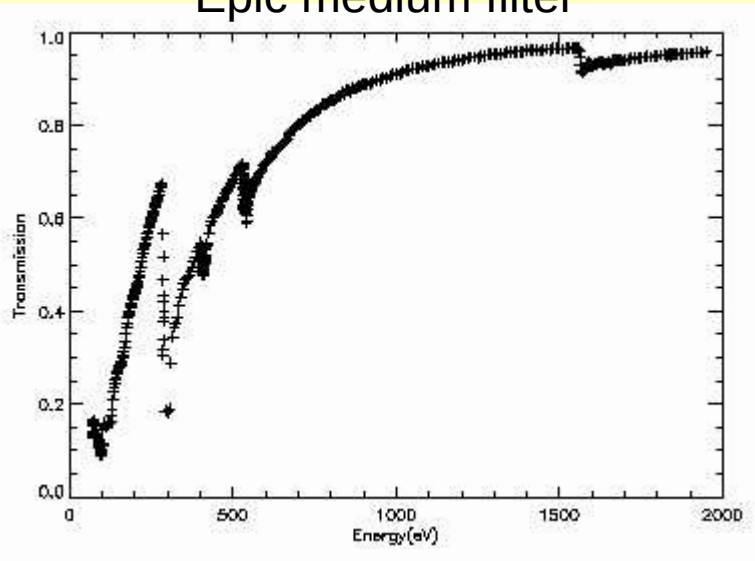




The vignetting of the MOS field of view includes an azimuthal component due to the gratings which capture ~50% of the light.

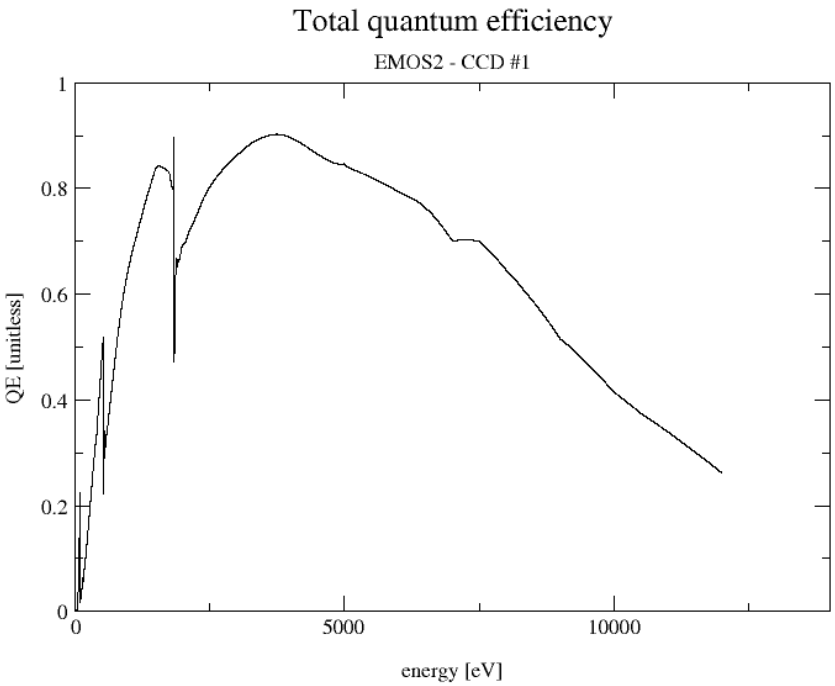
Filter Transmission

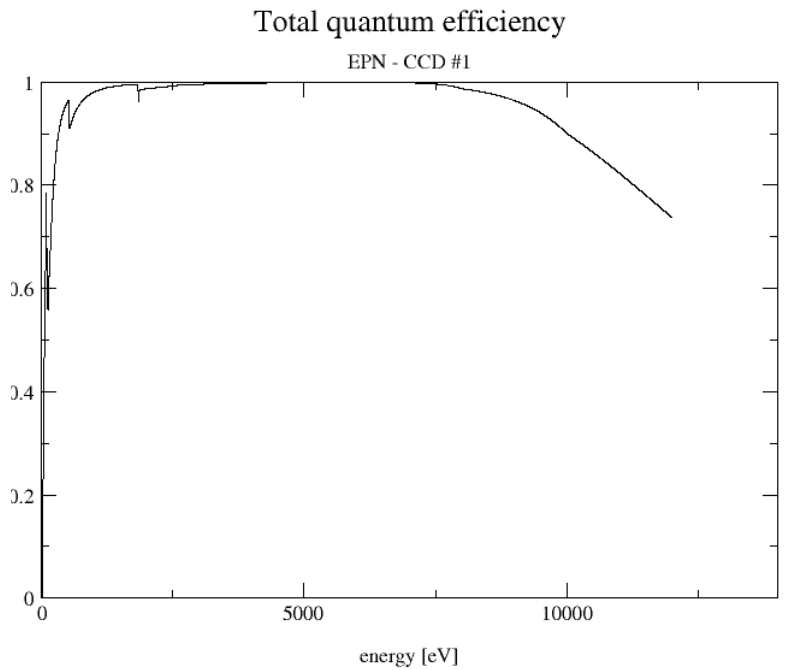
Epic medium filter



Option: *modelfiltertrans*=yes (default)

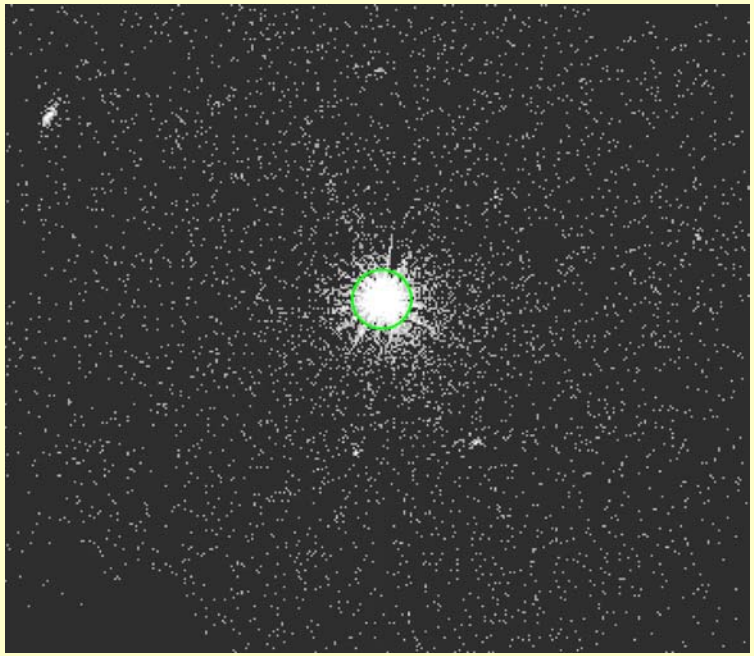
Detector Quantum Efficiency

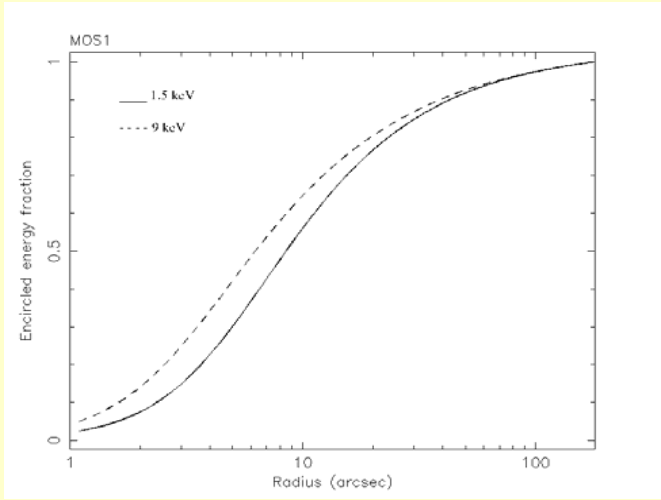




Option: *modelquantumeff*=yes (default)

Encircled Energy Correction



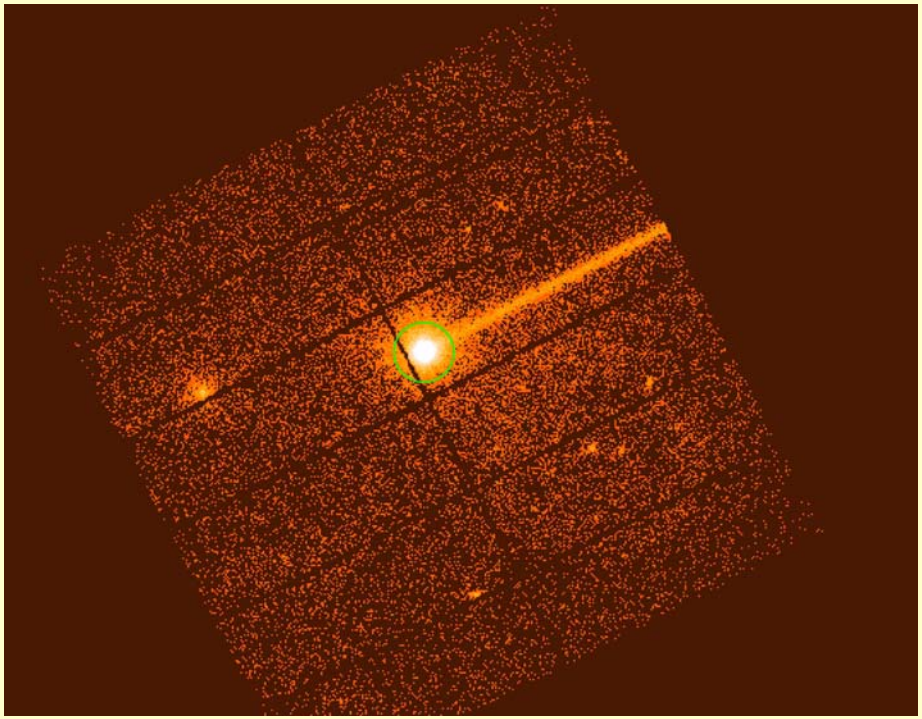


arfgen corrects for flux scattered out of the source extraction region.

Option: *modelee*=yes (default)

This is weakly dependent on energy and off-axis angle.

CCD gaps and Bad Pixels



arfgen corrects for the effective area lost due to chip gaps, bad pixels and offset columns.

Bad pixel and offset column information is stored in the event file header

Options: *withbadpixcorr*=yes (default) *badpixlocation*=myevents.fit

Point source: *arfgen* commands

Point source:

> arfgen spectrumset=spectrum.ds arfset=myspec.arf
badpixlocation=myevents.FIT detmaptype=psf

Using a canned matrix:

> arfgen spectrumset=spectrum.ds arfset=myspec.arf
badpixlocation=myevents.FIT detmaptype=psf withrmfset=yes
rmfset=***epn_ff20_sdY9.rmf***

Extended source: *arfgen* commands

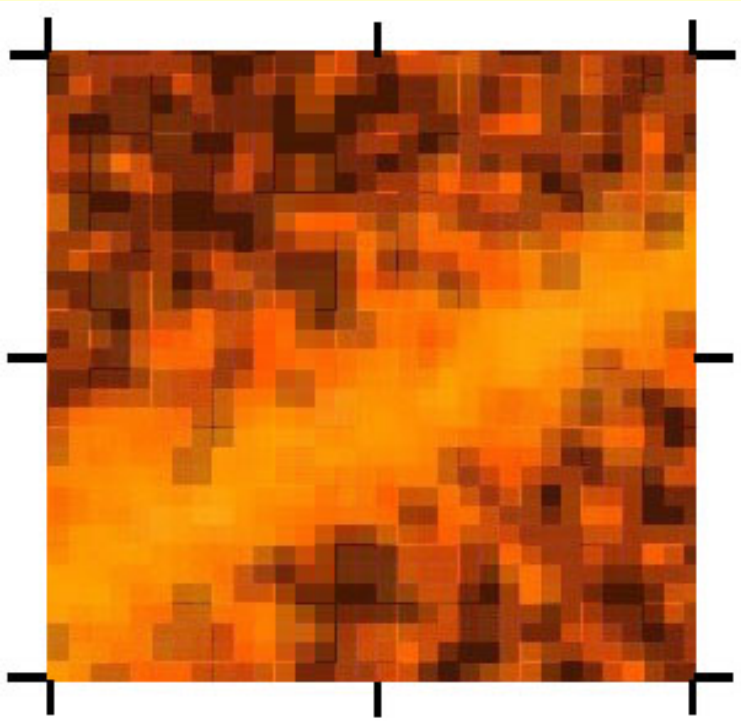
Extended source:

> arfgen spectrumset=spectrum.ds arfset=myspec.arf extendedsource=yes badpixlocation=myevents.FIT detmaptype=flat

Using a detector map:

> arfgen spectrumset=spectrum.ds arfset=myspec.arf extendedsource=yes badpixlocation=myevents.FIT detmaptype=dataset datamaparray=coarseimage.ds

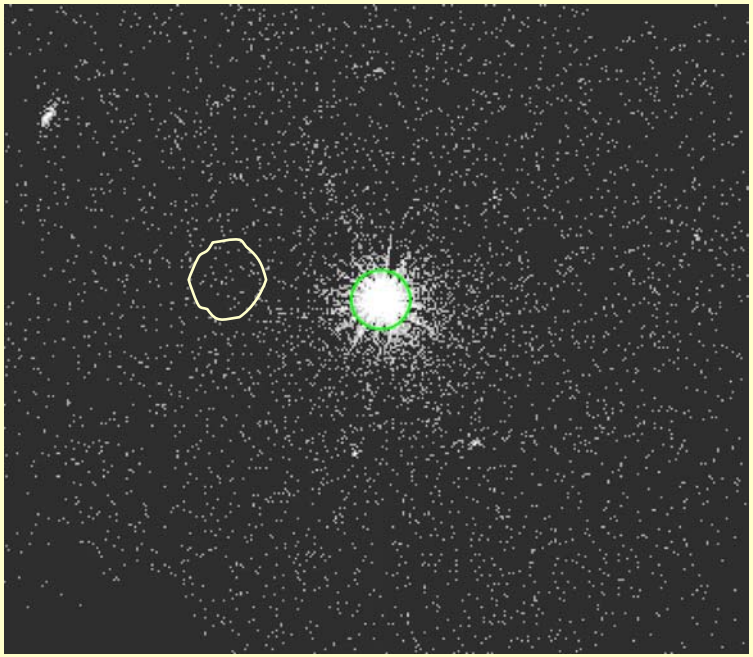
Extended source: detector map



- Create a coarsely binned image in detector coordinates**.**
- Run ***arfgen*** in extendedsource mode and flux-weight the ARF

arfgen spectrumset=spec.ds arfset=myspec.arf extendedsource=yes detmaptype=dataset detmaparray=coarseimage.ds

Influence of other sources



> arfgen spectrumset=cluster.ds
detmaptype=dataset
datamaparray=coarseimage.ds
crossregionarf=yes
crossreg_spectrumset=pointsource.ds

To calculate the contribution of flux from one region onto another region use the CROSSARF technique. The detector map must cover both regions and have at least 300 pixels within each area.

When should you use the canned matrices ?

For all instruments and observing modes, always use ***arfgen*** and either ***rmfgen*** or canned RMFs.

In general use ***rmfgen*** as it can be more accurate.
But…

*Canned RMFs can save time as **rmfgen** takes ~5 minutes to run.*

Canned RMFs may be issued between SAS releases to give access to the latest calibration.

Making life easy

Single task ***especget*** available:

- ✓ Takes source and background region
- ✓ Calculates centroid and optimum extraction radius
- ✓ Produces source and background spectra
- ✓ Generates appropriate ARF
- ✓ Optionally generates RMF
- ✓ Prepares files for spectral fitting

Use directly from ***xmmselect,*** ***"***OGIP spectral products***"***