SSC Pipeline Processing

Julian Osborne (Leicester University, julo@star.le.ac.uk)

November 22 2001

Abstract

A brief description of the pipeline processing that is performed by the SSC

1 Introduction

The SSC performs pipeline processing at Leicester of all ODFs sent by the SOC, returning them for distribution to the user

The aim of the pipeline is to eliminate the need for substantial user data processing, and to provide multiple entry points and views of the observation

The pipeline processes all science exposures, mostly as independent entities. The pipeline does not combine observations

Pipeline processing is fully automatic (but for a visual screening step)

The pipeline makes use of public SAS tasks, producing FITS and graphics files in compressed formats

2 System design

The pipeline consists of Perl modules which start as soon as their input data are ready

This non-sequential system (similar to that used for HST) is well suited to a multiple CPU environment

The modules invoke the SAS tasks which do the bulk processing

Many ODFs may be present in the system (at various stages of processing) at any time

ODFs and pipeline products are moved between the SOC & SSC via a dedicated line

3 Major steps

For each ODF the pipeline consists of the following major stages

- ODF ingest
- Exposure and GTI selection
- OM processing
- EPIC processing
- RGS processing
- EPIC source catalogue correlation
- HTML product generation
- Visual screening
- Compression, grouping & dispatch

4 SSC Pipeline Processing: 1 - ODF ingest

Initial actions in the pipeline processing of an ODF are:

- untar & uncompress ODF from XMM file transmission system
- check essential files present
- run *odfingest* to make enhanced ODF summary file
- enter ODF into pipeline database & move ODF to processing directory
- identify ODF to a specific pipeline (e.g. test or production)
- make observation summary html product & copy to INDEX.HTML
- define CCF set for this pipeline run
- make CALINDEX product

5 SSC Pipeline Processing: 2 - exposure selection & GTI creation

This step selects those exposures to be processed and creates good time intervals based upon attitude and, for the EPIC MOS, HK limits:

- check that ODF event lists exists for this exposure
- get filter wheel position & instrument mode
- ignore exposure if filter = Cal/Unknown/Blocked/Grism/Magnifier/Closed
- ignore exposure if mode = Undefined/HTR
- use *atthk* to make attitude timeseries product
- use *tabgtigen* to make attitude GTI file (att exists & att-median < 3 arcmin)
- use *hkgtigen* to make MOS HK GTIs (using HKPARMINT CCF file to select instrument subsystem settings, temperatures, thresholds)

6 SSC Pipeline Processing: 3a - OM exposure & image science window

- use *omflatgen* to get flatfield and *omprep* to put keywords in tracking files
- use *omthconv* to make tracking star intensity timeseries fits product
- use *omdrifthist* to make tracking history plot
- if Eng-2 use *omcomb* to reconstitute the full field image
- If SW in image mode:
  - use *omprep* to insert keywords into SW files
  - use *omcosflag* to apply tracking history to the bad pixel map
  - use *omflatfield* to apply tracking history to the flatfield and devide into image
  - use *ommodmap* to determine & apply event detection pattern correction
  - use *omdetect* to make multi-scale source list (extension of Sextractor)
  - use *ommag* to convert source intensities to instrumental magnitud
  - use *ommat* to add RA & Dec to source lists, and to rotate SW images to N up

7 SSC Pipeline Processing: 3b - OM fast mode & multi-exposure

If SW in fast mode:

- use *omprep* to insert keywords into SW files
- use *omfastshift* to apply tracking history to the 'event list'
- use *omfastflat* to make the flatfielded image
- use *omdetect* to make multi-scale source list (extension of Sextractor)
- use *ommat* to rotate SW images to North up
- per source:
  - use *omregion* to make circular source & annular background regions
  - use *evselect* to make the source and background timeseries
  - use *omlcbuild* to make timeseries fits file products
  - use *lcplot* to make timeseries plot product

8 SSC Pipeline Processing: 3c - OM observation-wide products

Use *omsrclistcomb* cross-identify OM sources from all independent windows, making an observation-wide source list with UBV standard magnitudes if colours are available

Use *ommosaic* to make multi-exposure, full field of view images for each filter and for each of the following OM setups:

- Rudi-5 (standard sequence of exposures to fill FOV, low resolution SWs only)
- engineering-2 (full field low resolution)
- engineering-4 (full field high resolution) - but not yet
- User defined (i.e. any other low resolution window configuration)

9 SSC Pipeline Processing: 4a - EPIC MOS processing

For each CCD event list:

- use *emframes* to allocate times to frames and make CCD GTI files
- if image mode, flag bad pixels using background flare screened data
- use *emevents* to perform full event resolution (use offset/var map if available)
- use *attcalc* to add projected celestial coordinates to event list
- use *emenergy* to add CCD-independent (PI) energies to event list
- use *evselect* to clean event list of bad events

For total event list:

- use *evlistcomb* to combine CCD events lists
- use *evselect* to add selection information to event list, copy CIF into event list
- use *evselect* to make timeseries of Gatti events (E>12 kev) from exterior CCDs
- use *tabgtigen* to make background flare GTI file from this timeseries

10 SSC Pipeline Processing: 4b - EPIC PN processing

For each CCD event list:

- use *epframes* to reformat event list, allocate frame times & make CCD GTI files
- use *bapixfind* to make mask for background events
- use *badpixfind* & *badpix* to flag bad pixels
- use *epevents* to perform event reconstruction and calibrate event energies
- use *attcalc* to add projected celestial coordinates to event list
- use *evselect* to clean event list of bad events (& PI<150 eV, remove edge pix)

For total event list:

- use *evlistcomb* to combine CCD events lists

- use *evselect* to make GTI filtered event list, add selection info & copy CIF
- use *evselect* to make timeseries of events (E=7-15 kev) from masked area
- use *tabgtigen* to make background flare GTI file from this timeseries

11 SSC Pipeline Processing: 4c - EPIC image & source list creation

For each energy band (0.2-0.5, 0.5-2.0,2.0-4.5,4.5-7.5,7.5-12 keV):

- use *evselect* to make 4" pixel sky coord fits image products (patt .le.12 for PN)
- use *implot* to make image plot products

For each EPIC exposure:

- use *eexpmap* to make exposure maps (all bands) & emask to limit detection area
- use *eboxdetect* on all band-limited images together to make box local source list
- use *esplinemap* to fit model background to source-excluded image
- use *eboxdetect* with model background to make box map source list
- use *emldetect* to make max likelihood source list
- use *evselect* to make image & *esensemap* to make 0.5-4.5 keV sensitivity map

Use *srcmerge* to make all-epic cross-identifed source list (fits & html)

Use *emosaic* to make full band all-epic image plot

12 SSC Pipeline Processing: 5a - RGS event list creation

Use *rgssources* to make list of sources >0.2 c/s & in RGS FOV from the appropriate MOS list (will use proposal position if none) For each CCD:

- use *rgsframes* to timestamp & quality flag CCD frames
- use *evselect* to remove bad frames and bad attitude intervals
- use *rgsbadpix* to identify known & new bad pixels
- use *rgsevents* to reconstruct and energy calibrate events

Use *rgsangles* to define event dispersion & cross-dispersion coordinates (based on prime source)

Use *rgsfilter* to make clean event list and exposure map products

13 SSC Pipeline Processing: 5b - RGS source products creation

Use *rgsregion* to make source extraction regions in (dispersion, X-dispersion) & (dispersion, PI energy) spaces

Use *evselect* to remove events PI < 170 eV, and to make (dispersion, energy) and (dispersion, X-dispersion) fits image products

Use *rgsimplot* to make plots of these, including extraction regions

For selected sources in RGS source list:

- use *rgsspectrum* to make background & background-subtracted source 1st & 2nd order spectrum fits products
- use *rgsspecplot* to make spectrum plot products

14 SSC Pipeline Processing: 6 - EPIC source catalogue correlation

The all-EPIC source lists is dispatched via secure ftp to OAS (Strasbourg) for catalogue cross-correlation, products return by the same means

Products are in fits, html or graphics formats and include:

- X-ray catalogue objects in the field of view
- EPIC source correlations in a large number of selected catalogues
- the list of catalogues searched
- descriptions of the fields of the catalogue extracts
- plot of catalogue entries on EPIC image
- plot of EPIC countours on Rosat All Sky Survey image
- DSS finding charts for each source

15 SSC Pipeline Processing: 7 - final stages

Essential keywords are inserted into all fits products to identify the revolution, various software versions and the product type

The product index file (listing all products) is made

The html summary products are made by *ppssumm*

The products are distributed by secure ftp to SSC sites for screening, this consists of:

- completeness check
- visual examination of plot products
- examination of fits images together with EPIC source lists
- brief report

The products are compressed & grouped, then sent via XMM file transmission system to SOC

16 SSC Pipeline Processing: more info

Since 20 Dec 2000 there have been 8 pipeline revisions, many incorporating pipeline-specific SAS revisions

I have described the pipeline revision expected to become operational about now, there will be a further significant revision when the next public SAS release occurs

Summary & detailed release notes for all pipeline versions, as well as pipeline module code and software version number information, is all available from:

http://xmmssc-www.star.le.ac.uk/pipeview/prod

References