rgsprods

August 16, 2005

Abstract

This is a perl task that creates XMM RGS data products not made by **rgsproc**.

1 Instruments/Modes

Instrument	Mode
RGS	spectroscopy

2 Use

pipeline processing	no
interactive analysis	yes

3 Description

There are two ways to make XMM products for the RGS cameras: via the XMM data pipeline or via the SAS task **rgsproc**. The latter does not however produce the full set of product files: the images (both FITS and PNG) are not produced, nor are the PDF spectrum plots. It is the function of **rgsprods** to supply this lack - ie, the sum of the RGS pipeline functionality is intended to be duplicated by **rgsproc** followed by **rgsprods**.

rgsprods should be run from the directory in which the products are stored.

4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
astest	no	boolean	no	

Run in test mode (no invocation of tasks).

harsh	no	boolean	no	

When this is set, the task itself fails the first time any of its invoked tasks fail; whereas if `harsh`=no, the task attempts to keep going if at all possible.

5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

noMatchingFiles *(error)*
No useable product files were found in the PWD.

noEvlist *(error)*
The task can't find an RGS event list with the correct XMM-product name format in the PWD.

evselectFailed *(error)*
The task attempted to invoke **evselect** but could not. Note that if `harsh`='no', a warning rather than an error results.

noFitsDetImage *(error)*
The task is ready to convert the detector-plane FITS image into GIF form, but the FITS image cannot be found. Note that if `harsh`='no', a warning rather than an error results.

rgsimplotFailed *(error)*
The task attempted to invoke **rgsimplot** but could not. Note that if `harsh`='no', a warning rather than an error results.

noGifDetImage *(error)*
The task is ready to convert the detector-plane GIF image into PNG form, but the GIF image cannot be found. Note that if `harsh`='no', a warning rather than an error results.

giftopngFailed *(error)*
The task attempted to invoke the utility 'giftopng' but could not. Note that if `harsh`='no', a warning rather than an error results.

rmFailed *(error)*
The task attempted to invoke the shell function 'rm' but could not. Note that if `harsh`='no', a warning rather than an error results.

noFitsBananaImage *(error)*
The task is ready to convert the beta-PI (sometimes known as the banana plot) FITS image into GIF form, but the FITS image cannot be found. Note that if `harsh`='no', a warning rather than an error results.

noGifBananaImage *(error)*
The task is ready to convert the beta-PI (sometimes known as the banana plot) GIF image into PNG form, but the GIF image cannot be found. Note that if `harsh`='no', a warning rather than an error results.

rgsspecplotFailed *(error)*
The task attempted to invoke **rgsspecplot** but could not. Note that if `harsh`='no', a warning rather than an error results.

pstopdfFailed *(error)*
The task attempted to invoke the utility 'ps2pdf' but could not. Note that if `harsh`='no', a warning rather than an error results.

testFlagIsSet *(warning)*
The user has set `astest`='yes'. This mode is useful for testing the functionality of the perl code.
corrective action: No system calls are made, no files are written.

evselectFailed *(warning)*
The task attempted to invoke **evselect** but could not. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

noFitsDetImage *(warning)*
The task is ready to convert the detector-plane FITS image into GIF form, but the FITS image cannot be found. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

rgsimplotFailed *(warning)*
The task attempted to invoke **rgsimplot** but could not. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

noGifDetImage *(warning)*
The task is ready to convert the detector-plane GIF image into PNG form, but the GIF image cannot be found. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

giftopngFailed *(warning)*
The task attempted to invoke the utility 'giftopng' but could not. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

rmFailed *(warning)*
The task attempted to invoke the shell function 'rm' but could not. Note that if `harsh`='no', a warning rather than an error results.
corrective action: The task passes on to the next product.

noFitsBananaImage *(warning)*
The task is ready to convert the beta-PI (sometimes known as the banana plot) FITS image into GIF form, but the FITS image cannot be found. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

noGifBananaImage *(warning)*
The task is ready to convert the beta-PI (sometimes known as the banana plot) GIF image into PNG form, but the GIF image cannot be found. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

rgsspecplotFailed *(warning)*
The task attempted to invoke **rgsspecplot** but could not. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

pstopdfFailed *(warning)*
The task attempted to invoke the utility 'ps2pdf' but could not. Note that if `harsh`='yes', an error rather than a warning results.
corrective action: The task passes on to the next product.

6 XMM product format

XMM products adhere to a standard naming scheme (see [1]) which may be summarised by the following template:

```
Poooooooooooiiteeennnnnnddddd.xxx
```

The fields have the following meanings:

- oooooooooo: 10-digit observation ID.
- ii: instrument identifier (eg R1, R2).
- t: exposure type (usually either 'S' or 'U', for 'scheduled' or 'unscheduled').
- eee: 3-digit exposure ID.
- nnnnnn: 6-character product type identifier.
- dddd: 4-digit utility number.
- xxx: file data type identifier. Examples are 'FIT', 'GIF', 'PNG', 'PDF'.

7 Input Files

1. Calibrated RGS event list; nnnnnndddd.xxx = 'EVENLI0000.FIT'.
2. RGS source list dataset, with associated region extensions; nnnnnndddd.xxx = 'SRCLI_0000.FIT'.
3. RGS spectra; nnnnnndddd.xxx = 'SRSPECdddd.FIT'. The first digit of 'dddd' should be the spectral order number and the last three digits the row number of the source in the RGS source list.

8 Output Files

Products as follows:

1. FITS version of the detector-plane image. nnnnnndddd.xxx = 'IMAGE_0000.FIT'.
2. PNG version of the detector-plane image. nnnnnndddd.xxx = 'IMAGE_0000.PNG'.
3. FITS version of the beta-PI ('banana') image. nnnnnndddd.xxx = 'ORDIMG0000.FIT'.
4. PNG version of the beta-PI ('banana') image. nnnnnndddd.xxx = 'ORDIMG0000.PNG'.
5. For each source for which there is a spectrum (see section 7), a PDF spectrum plot of all orders. nnnnnndddd.xxx = 'SRSPEC0ddd.PDF' where the 3 digits ddd are the same as the corresponding digits of the input spectrum files.

9 Algorithm

```
run evselect to make banana_fits
run rgsimplot to make banana_gif
giftopnm banana_gif | pnmtopng > banana.PNG

run evselect to make detplane_fits
run rgsimplot to make detplane_gif
giftopnm detplane_gif | pnmtopng > detplane.PNG

foreach (source) {
  run rgsspecplot with list of fits spectra as input, to make PDF spectrum plot
} end foreach source
```

10 Comments

-

References

[1] Julian Osborne. Data products specifications. Technical Report SSC-LUX-SP-0004, SSC, 2000.