



ecoordconv

February 1, 2016

Abstract

A routine to convert a position between coordinate systems

1 Instruments/Modes

Instrument	Mode
EPIC	IMAGING, TIMING

2 Use

pipeline processing	no
interactive analysis	yes

3 Description

This task converts a position in an image from one coordinate system into positions in other coordinate systems. For example the routine can be used to convert a celestial sky position (RA, DEC) into detector coordinates (DETX, DETY), chip coordinates (RAWX, RAWY), X/Y pixels and TelCoords (theta, phi). It also gives the numbers of the CCDs which are included within an input region.

3.1 Input

3.1.1 Source image

The source image may be input in detector or sky (X/Y) coordinates. The task expects to find astrometry keywords in a certain part of the primary header and will exit with an error if the keywords are not found. Standard images produced by `evselect` and `xmmselect` and the pipeline will process ok. The background spline maps produced by the pipeline source detection chain and the exposure maps produced by `eexppmap` need to be pre-processed before they can be used within `ecoordconv` (see section 8).



Table 1: Output quantities

Value	Units	Units
Theta	Off-axis angle	arc seconds
Phi	Azimuthal angle	Radians
X	X sky coord	0.05 arcsec pixel
Y	Y sky coord	0.05 arcsec pixel
DETX	Detector X coord	0.05 arcsec pixel
DETY	Detector Y coord	0.05 arcsec pixel
RA	Right ascension	degrees
DEC	Declination	degrees
RAWX	X chip coordinate	pixel
RAWY	Y chip coordinate	pixel
CCD(s)	CCD number(s) in region	-
Central CCD	CCD at region centre	-

3.1.2 Regions

Spatial regions may be entered in raw chip, detector, sky (X/Y) or celestial (RA, DEC, FK4 2000) coordinates. If raw coordinates are used the CCD must be given using the parameter `ccdno`. If the coordinates of the region are not the same as those of the image, e.g. a sky pixel region on a detector coordinate image, then the task will still function but will run more slowly. An error will result if the spatial region is not within the image. Any shape conforming to the **selectlib** rules may be used, except that RAW coordinates may currently only be entered using a circular selection.

3.2 Output

The output text shown below is independent of the SAS_VERBOSITY setting. The strings shown may be searched for in a script and every effort will be made to keep them constant between versions of this task.

```

ecoordconv:- Region Centre:
Theta: Phi: 105.768 2.17488
X: Y: -1239.05 1711.11
DETX: DETY: -1239.05 1711.11
RA: DEC: 275.547 64.3216
RAWX: RAWY: 54 167
CCD(s): 1 2 4 5 7 8 10 11 centred on CCD: 4

```

3.3 Examples

1. Convert a particular sky coordinate position

```
ecoordconv imageset=pnimagexy.ds x=27000 y=26900 coordtype=POS
```

```

ecoordconv:- Region Centre:
Theta: Phi: 18.4712 2.59867
X: Y: 27010 26888

```



DETX: DETY: -353.754 160.874
RA: DEC: 275.505 64.3385
RAWX: RAWY: 43 186
CCD(s): 4 centred on CCD: 4

2. Convert a detector coordinate region

ecoordconv srcexp="(DETX,DETY) in CIRCLE(100,1000,7000)" imageset=pnimagexy.ds

ecoordconv:- Region Centre:
Theta: Phi: 50.5951 1.40977
X: Y: 26280.112 27503.672
DETX: DETY: 124.823 968.882
RA: DEC: 275.498 64.3258
RAWX: RAWY: 37 176
CCD(s): 1 4 5 7 8 10 centred on CCD: 4

3. Convert a RAW coordinate

ecoordconv imageset=image.ds coordtype=raw x=32 y=191 ccdno=4

ecoordconv:- Region Centre:
Theta: Phi: 48.3652 3.84547
X: Y: 27095.5 27965.5
DETX: DETY: 546.171 -243.231
RA: DEC: 114.952 -85.6571
RAWX: RAWY: 32 191
CCD(s): 4 centred on CCD: 4

4. Convert from sky to celestial coordinates only

ecoordconv imageset=image.ds coordtype=pos x=25500 y=26200 pos2eqpos=yes

ecoordconv:- Region Centre:
RA: DEC: 115.012 -85.6611

4 Parameters

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

Table with 5 columns: Parameter, Mand, Type, Default, Constraints

Table row for parameter 'imageset' with values: yes, string, image

The name of the input image.

Table row for parameter 'srcexp' with values: no, string

Expression for defining the source position and parameters



withcoords	no	boolean	false	none
-------------------	----	---------	-------	------

If **true**, the source position must be specified via the parameters **coordtype**, **x**, **y**. Otherwise, the source position is taken from the centre of the source region defined in the **srcexp** parameter.

x	no	real		
----------	----	------	--	--

The X coordinate of the position to convert

y	no	real		
----------	----	------	--	--

The Y coordinate of the position to convert

coordtype	yes	string	eqpos	eqpos pos det raw
------------------	-----	--------	-------	-------------------

The coordinate system for which the source position, specified by the parameter **x** and **y**, is defined. If **coordtype** is set to **eqpos**, then **x**, **y** correspond to RA and DEC respectively in decimal degrees. If **coordtype** = **pos**, then **x**, **y** correspond to POS coordinates. (Note that the POS coordinates are defined relative to a nominal pointing position; this is taken from the global attributes **REFXCRVL** and **REFYCRVL** of the image dataset.) If **coordtype** = **det**, then **x**, **y** are the x and y positions of the source centre, in DET coordinates. Finally, if **coordtype** = **raw**, then **x**, **y** are the raw chip positions of the source centre. In this case the CCD number must be specified using the **ccdno** parameter.

pos2eqpos	no	boolean	false	none
------------------	----	---------	-------	------

If **true**, then only the conversion from x/y to ra/dec coordinates is made. This option is only active when **withcoords**=**true** and **coordtype**=**pos**. It avoids problems which occur when the RA_PNT, DEC_PNT coordinates are more than 90 degrees away from the X/Y reference pixel position.

withccd	no	boolean	false	none
----------------	----	---------	-------	------

If **true**, the ccd number must be specified via the parameter **ccdno**, **x**, **y**. This is mandatory if the position has been specified in raw chip coordinates.

ccdno	yes	int		
--------------	-----	-----	--	--

The CCD number on which the RAW chip position falls.

theta_out	no	double		
------------------	----	--------	--	--

Output parameter that contains the off-axis angle, in units of arc seconds.

phi_out	no	double		
----------------	----	--------	--	--

Output parameter that contains the azimuthal angle, in units of radians.

ra_out	no	double		
---------------	----	--------	--	--

Output parameter that contains the right ascension.

dec_out	no	double		
----------------	----	--------	--	--

Output parameter that contains the declination.

posx_out	no	double		
-----------------	----	--------	--	--

Output parameter that contains the POS, X position.

posy_out	no	double		
-----------------	----	--------	--	--

Output parameter that contains the POS, Y position.

detx_out	no	double		
-----------------	----	--------	--	--

Output parameter that contains the X position in detector coordinates

dety_out	no	double		
-----------------	----	--------	--	--



Output parameter that contains the Y position in detector coordinates

rawx_out	no	double		
-----------------	----	--------	--	--

Output parameter that contains the X position in raw, chip coordinates

rawy_out	no	double		
-----------------	----	--------	--	--

Output parameter that contains the Y position in raw, chip coordinates

ccd_out	no	int		
----------------	----	-----	--	--

Output parameter that contains the number of the CCD which the input position falls on.

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.

InvalidExpression (*error*)

The input expression for this spatial region was not valid.

InvalidRegion (*error*)

The centre of the source or background region lies outside of the image

invalidArraySize (*error*)

The image is not two dimensional.

InvalidWCSType (*error*)

The image axes specified in the CTYPE1 and CTYPE2 keywords are not compatible.

invalidCoordType (*error*)

The input coordinate system is not recognised.

noCCDNumber (*error*)

A position has been given in RAW chip coordinates but no CCD number has been supplied. Use `ccdno=nn` on the command line.

UnknownModeString (*warning*)

If the spectrum contains an observing mode (in the keyword SUBMODE) which is not recognised then the software assumes that the common PrimeFullWindow mode was in use.

corrective action: PrimeFullWindow

NoInstrument (*warning*)

If the INSTRUME keyword is not set in the image header a warning is issued and the default of MOS-1 is taken. The instrument determines the PSF used and the position of the optical-axis.

corrective action:

InvalidPosWCSInfo (*warning*)

The REFERENCE keywords in the image header, e.g. REF_XCRPX, REF_XCRVL, REF_XCDLT are incomplete. Defaults are chosen but there is likely to be a problem later.

corrective action:



6 Input Files

- an EPIC image produced by `evselect` or `xmmselect` or the pipeline.

7 Output Files

8 Algorithm

9 Comments

The routine now gives the *exact* translation between coordinate systems. It no longer returns the coordinates of the centre of the nearest image pixel.

References