Chandra X-ray Center

AHELP for CIAO 3.4   **dmcoords**   Context: tools

Jump to: Description Examples Parameters CHANGES IN CIAO 3.0 Bugs See Also

Synopsis

Convert between Chandra coordinate systems

Syntax

```
dmcoords infile [asolfile] [options]
```

Description

The dmcoords tool allows you to convert between the various coordinate systems used by Chandra (eg SKY to CHIP) and to predict the source location for a variety of instrumental configurations. It has limited support for data from other telescopes.

The tool has two modes of operation:

- parameter mode – where all the input and output options are set using the parameter interface.
- interactive mode – in which commands can be entered at a prompt and the output is to the screen.

The parameter mode is particularly useful in scripts since it can be run with no user interaction, whereas the interactive mode allows greater flexibility and the ability to perform multiple conversions in one session.

Setting the instrument configuration

The coordinate converion depends upon the instrument configuration of Chandra (for instance, which instrument is in the focal plane). If you already have data for an observation then this information is available in the header of the event file (and files created from this one, such as images) and you can use the infile parameter to point to such a file.

However, if you do not have such a file (for instance you are planning an observation and want to know whether a particular source will fall on a chip) then you can set the various values using either the configuration parameters (detector, grating, fpsys, sim, displace, ra_nom, dec_nom, roll_nom, ra_asp, dec_asp, and roll_asp), or the SET command if in interactive mode. Note that settting the values in this manner overrides the values read in from the configuration file (i.e. the infile parameter). One time that this comes in handy is when you want to use a different aspect solution, as described below.

Coordinate conversions and the aspect solution

Most of the coordinate conversions (e.g. SKY to CHIP) depend upon the aspect solution. This means that the conversion is time dependent due to the dither performed by Chandra during an observation. The dmcoords tool will perform the conversion using a single (ie instantaneous) aspect solution. Unless otherwise specified (i.e. using either the ra_asp, dec_asp, and roll_asp parameters or the SET ASPECT command) the nominal aspect solution from the configuration file is used, as given by the RA_NOM, DEC_NOM, and ROLL_NOM keywords.

Note that the default amplitudes of the Chandra dither motion are 16 and 40 arcseconds for ACIS and HRC observations respectively. The actual aspect solution information for a given observation is stored in the set of pcad*asol1.fits files.

Parameter mode

To convert from a given position, set the option parameter to the name of the input coordinate system (e.g. option=sky) and the corresponding set of coordinate parameters (e.g. x and y for option=sky) to the input position. After running dmcoords the remaining coordinate parameters (e.g. chip_id, chipx, chipy, tdetx, ...) will have been set. The pget tool can then be used to access this information:

```
unix% pget dmcoords chipx
662.0891847226883
```

When using the parameter mode it is recommended that the verbose parameter be set to 1 if you wish to see the output values (they will always be stored in the parameter value, whatever the value of verbose). The default setting for the verbose parameter (0) is intended for pipeline/batch use when screen output is undesireable.

Parameters for the various coordinate systems:

Option value	Parameters
cel	ra, dec
sky	x, y
det	detx, dety
chip	chip_id, chipx, chipy
logical	logicalx, logicaly
msc	theta, phi

Interactive Mode

The interactive mode of dmcoords can be entered by not specifying a value for the option parameter. The prompt will change to 'dmcoords>:' and commands can then be entered to set or query the instrument configuration and to perform coordinate conversions. Unlike the parameter mode, more than one conversion may be calculated by dmcoords within a single session. Note that the commands are case–insensitive.

Interactive mode commands

Command	Notes
HELP	Lists the available commands
Q (or QUIT)	Exits the program
STATUS	Displays the current instrumental configuration
SET ?	Displays the configuration options
SKY x y	Converts sky location

CEL ra dec	Converts celestial location
SET CHIP ACIS−S2	Sets the chip (needed by the CHIP command)
CHIP chipx chipy	Converts chip location
DET detx dety	Converts detector location
DEG / HMS	Toggle between degree and sexagesimal format

Note that in interactive mode the "SET NOMINAL" command requires that the input be in sexagesimal format, whereas "SET ASPECT" uses the setting of the celfmt parameter to decode the supplied RA and DEC values. Use the "DEG" and "HMS" interactive commands to switch between formats.

Example 1

```
dmcoords acis.fits option=cel ra=06:23:41 dec=-00:02:31 verbose=1
dmcoords acis.fits option=cel celfmt=deg ra=95.92083 dec=-0.04194
verbose=1
```

These two equivalent uses read the header of acis.fits and print out (and write to the parameter file) values of sky, detector and chip coordinates corresponding to the given position. The celfmt paramer controls the format used (sexagesimal or degrees−hms/deg).

The pget command can be used to access the values of the transformation: for example to get the x value of the sky coordinate system one could use

```
   unix% pget dmcoords x
```

Example 2

```
dmcoords acis.fits opt=chip chip_id=7 chipx=963.0 chipy=512.0
```

This takes a chip position (including the number of the chip) and calculates the location in the other coordinate systems.

Example 3

```
dmcoords acis.fits
```

Since no option parameter was specified, this command puts you into the interactive version of the program. The prompt will change to 'dmcoords:>'; use the HELP command (case insensitive) to list the available commands.

To repeat the calculation from example 2, one would enter:

```
   dmcoords:> set chip acis-7
   dmcoords:> chip 963 512
```

Example 4

```
dmcoords none opt=det detx=4963.0 dety=4012.0 sim="0.6 0.0 -192.3"
detector=acis celfmt=deg ra_nom=96.1043 dec_nom=-0.0243 roll_nom=212.3
verbose=1
```

Instead of using an input file to set up the header, here we specify the configuration parameters explicitly.

Example 1   3

The output of the command looks like:

```
---------------------------------------
Spacecraft Configuration:
---------------------------------------
Dataset:
Observatory:
Telescope:         CHANDRA
Grating:           NONE
Instrument:        ACIS
Detector:          ACIS
Chip:              ACIS-S4
SIM position:          0.600      0.000   -192.300
SIM offset:            0.000      0.000      0.000
SIM rotation:          0.000      0.000      0.000
Grating order: 0
Focal length:      10070.000000
Chip pixel scale: 0.023987 x 0.023987 mm/pixel
Chip size:         1024 x 1024 pixels
---------------------------------------
Sky Coordinate Configuration:
---------------------------------------
Sky coordinate pixel system: FP-1.1
Center pixel of sky plane:  (4096.500000,4096.500000)
Size of sky plane:          8192 x 8192
Sky pixel scale:            0.492000 arcsec/pixel
Nominal:        RA  96.104300 Dec  -0.024300 Roll   0.000000
Aspect:         RA  96.104300 Dec  -0.024300 Roll 212.300000
--------
(RA,Dec):      06:24:47.574    +00:02:55.46
(RA,Dec):        96.19823         0.04874 deg
THETA,PHI         7.139'          354.43 deg
(Logical):       3409.23         4630.94
SKY(X,Y):        3409.23         4630.94
DETX,DETY        4963.00         4012.00
CHIP ACIS-S4       46.58          706.29
TDET             5005.58         2408.29
```

Parameters

name	type	ftype	def	min	max	units	reqd	stacks
infile	file	input					yes	
asolfile	file	input					no	yes
option	string							
chip_id	integer							
chipx	real					pixels		
chipy	real					pixels		
tdetx	real					pixels		
tdety	real					pixels		
detx	real					pixels		
dety	real					pixels		
x	real					pixels		
y	real					pixels		
logicalx	real					pixels		
logicaly	real					pixels		

ra	string							
dec	string							
theta	real			0	10800	arcmin		
phi	real			0	360	degrees		
order	integer		0					
energy	real		1.0			keV		
wavelength	real		0			Angstrom		
ra_zo	string							
dec_zo	string							
celfmt	string		hms					
detector	string							
grating	string							
fpsys	string							
sim	string							
displace	string							
ra_nom	string							
dec_nom	string							
roll_nom	string					degrees		
ra_asp	string							
dec_asp	string							
roll_asp	string					degrees		
geompar	file		geom					
verbose	integer		0	0	5			

Detailed Parameter Descriptions

Parameter=infile `(file required filetype=input)`

Input file, or dm defined virtual file

dmcoords uses the file header to initialize the configuration. You can enter 'none' and set up the configuration from the parameters.

Parameter=asolfile `(file not required filetype=input stacks=yes)`

Aspect solution file(s)

dmcoords uses the aspect solution file(s) to get the aspect and sim offsets by calculating the mean of the dy column, the dz column, and the dtheta column.

Parameter=option `(string)`

Which coordinate system to start from (cel/sky/det/chip/logical/msc).

The option parameter can have one of the following values:

- cel: specifies that ra and dec are the parameters given on input. The format assumed for ra and dec is specified by the celfmt parameter, see below.
- sky: the input parameters are x and y, the sky pixel coordinate values (`physical' coordinates in the language of ds9 if you have made a sky image).
- det: the input parameters are detx and dety, the focal plane pixel coordinate values.
- chip: the input parameters are chip_id, chipx and chipy.
- logical: the input parameters are image logical coordinates, i.e. the binned sky pixel coordinates. Only relevant if the input file is an image rather than an event list.
- msc: the input parameters are mirror spherical coordinates; theta is the off axis angle in arcmin and phi is the azimuth in degrees.

Parameter=chip_id `(integer)`

The chip ID number; for ACIS, this lies between 0 and 9, for HRC between 0 and 3. Can be input or output.

Parameter=chipx `(real units=pixels)`

The chip X pixel coordinate. See the CXC Coordinates papers for definitions. Can be input or output.

Parameter=chipy `(real units=pixels)`

The chip Y pixel coordinate. See the CXC Coordinates papers for definitions. Can be input or output.

Parameter=tdetx `(real units=pixels)`

The tiled detector X pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on a fictitous plane on which each chip of the instrument is flattened out. Output only.

Parameter=tdety `(real units=pixels)`

The tiled detector Y pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on a fictitous plane on which each chip of the instrument is flattened out. Output only.

Parameter=detx `(real units=pixels)`

The focal plane X pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on the tangent plane to the mirror optical axis. Pixel size is fixed in angular size. Center of the image plane is the mirror axis. Can be input or output.

Parameter=dety `(real units=pixels)`

The focal plane Y pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on the tangent plane to the mirror optical axis. Pixel size is fixed in angular size. Center of the image plane is the mirror axis. Can be input or output.

Parameter=x `(real units=pixels)`

The sky plane X pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on the tangent plane to the nominal celestial pointing direction. Pixel size is fixed in angular size. Center of the image plane is RA_NOM, DEC_NOM. Can be input or output.

Parameter=y `(real units=pixels)`

The sky plane Y pixel coordinate. See the CXC Coordinates papers for definitions. Gives position on the tangent plane to the nominal celestial pointing direction. Pixel size is fixed in angular size. Center of the image plane is RA_NOM, DEC_NOM. Can be input or output.

Parameter=logicalx `(real units=pixels)`

The sky plane logical X pixel coordinate, for a binned image. If you took the event list and binned by a factor of 32, say, then original sky pixels 1 to 8192 correspond to logical pixels 1 to 256. Can be input or output.

Parameter=logicaly `(real units=pixels)`

The sky plane logical Y pixel coordinate, for a binned image. If you took the event list and binned by a factor of 32, say, then original sky pixels 1 to 8192 correspond to logical pixels 1 to 256. Can be input or output.

Parameter=ra `(string)`

RA and Dec are the right ascension and declination corresponding to sky pixels X,Y. This is ra in sexagesimal format (if celfmt=hms) or in degrees (if celfmt=deg). Can be input or output.

Parameter=dec `(string)`

RA and Dec are the right ascension and declination corresponding to sky pixels X,Y. If the celfmt parameter is `hms' use sexagesimal format, for example "06:23:11.21"; if the celfmt parameter is `deg' use decimal degrees, for example "95.79671". Can be input or output.

Parameter=theta `(real min=0 max=10800 units=arcmin)`

The off axis angle coordinate in arcmin. See the CXC Coordinates papers for definitions. Can be input or output.

Parameter=phi `(real min=0 max=360 units=degrees)`

The mirror spherical coordinate azimuth in degrees. See the CXC Coordinates papers for definitions. Theta and phi are polar coordinates whose tangent plane is given by detx and dety. Can be input or output.

Parameter=order `(integer default=0)`

The grating order, zero or positive or negative integer.

Parameter=energy `(real default=1.0 units=keV)`

The energy in keV. Only relevant for grating observations.

Parameter=wavelength `(real default=0 units=Angstrom)`

The wavelength in Angstroms. Only relevant for grating observations.

Parameter=ra_zo `(string)`

The RA of the zero order image, for grating observations.

Parameter=dec_zo `(string)`

The Dec of the zero order image, for grating observations.

Parameter=celfmt `(string default=hms)`

Celestial position format (hms/deg) i.e. sexagesimal or decimal−degree formats will be used for the RA,dec parameter pairs.

Parameter=detector `(string)`

Detector name (ACIS/HRC−I/HRC−S). If present, overrides the INSTRUME keyword in the input file.

Parameter=grating `(string)`

Grating name (NONE/LETG/HETG). If present, overrides the GRATING keyword in the input file.

Parameter=fpsys `(string)`

Focal plane coordinate system. ASC−FP−1.1 for ACIS; ASC−FP−2.1 for HRC. But see coordinates papers for more detail.

Parameter=sim `(string)`

Optical bench (SIM) position (in X,Y,Z), three real values separated by spaces. e.g. "0.0 0.0 −190.1". Overrides the SIM_X SIM_Y SIM_Z header values in the input file.

Parameter=displace `(string)`

Optical bench misalignment, six space−separated real values corresponding to a translation and rotation of the STF origin with respect to the FC origin. For expert users only.

Parameter=ra_nom `(string)`

Nominal pointing direction, in format specified by celfmt. This overrides the input file RA_NOM keyword.

Parameter=dec_nom `(string)`

Nominal pointing direction, in format specified by celfmt. This overrides the input file DEC_NOM keyword.

Parameter=roll_nom `(string units=degrees)`

Nominal roll angle, in degrees. This overrides the input file ROLL_NOM keyword.

Parameter=ra_asp `(string)`

Instantaneous pointing direction, in format specifies by celfmt. If omitted, ra_nom is used.

Parameter=dec_asp `(string)`

Instantaneous pointing direction, in format specifies by celfmt. If omitted, dec_nom is used.

Parameter=roll_asp `(string units=degrees)`

Instantaneous roll angle, in degrees. Overrides the roll_nom parameter.

Parameter=geompar `(file default=geom)`

The name of the Pixlib Geometry parameter file.

Parameter=verbose `(integer default=0 min=0 max=5)`

Verbose can be from 0 to 5, generating different amounts of debugging output. Remember to set verbose=1 if you want to see the standard information.

CHANGES IN CIAO 3.0

Fixed exit status on error. If infile='', will get all parameters from dmcoord's own .par file. Tool will now operate when infile does not contain WCS and opt=sky|cel (eg. infile=pcad*asol1.fits).

Bugs

Reprojected event files

dmcoords does not correctly calculate CHIP coordinates on reprojected event files. A workaround is listed on the dmcoords bugs page.

See the bugs page for this tool on the CIAO website for an up−to−date listing of known bugs.

See Also

tools

aconvolve, acrosscorr, arestore, dmfilth, dmregrid, mkpsf, psf_project_ray, reproject_events, reproject_image, reproject_image_grid, skyfov

---

Parameter=roll_asp (string units=degrees)