Chandra X-ray Center

AHELP for CIAO 3.4

dmtype2split

Context: tools

Jump to: Description Examples Parameters Bugs See Also

Synopsis

Create a type 1 output file for specified rows of a type 2 file.

Syntax

```
dmtype2split infile outfile [rows] [clobber] [verbose]
```

Description

CIAO extends the concept of type 1 and type 2 PHA files to general tables. A type 1 file of size N is a simple table consisting of scalar columns each with N rows. A type 2 file of size N x M is a table with M rows in which each column is either scalar or an array of size N. Each row of the type 2 file corresponds to a whole type 1 file, and `dmtype2split' performs this mapping.

The mapping consists of turning each array column entry from `sideways' to `vertical', so that an array of 4096 values becomes a column with 4096 rows. Scalar columns map to keywords in the output file.

Each of the rows in the type 2 file specified by either the "rows" parameter (whose use is currently being 'deprecated') or the datamodel filter syntax (i.e. infile="filname.fits[#row=N:M,P:Q,..]"), is copied to a type 1 file. The rows are specified in a comma– and colon–delimited list. Each output file has the basic header keywords and subspace copied from the input. If the input file is a pha file, keywords required by the OGIP format are also written; otherwise, generic keywords are written. For each scalar column in the input, the coordinate keyword information is written, and the datatype, value, and unit are written as a keyword. For each input column that is an array, a column is created from the datatype, value, and unit; the row of input data is read and written to the output column.

The rows can be specified in the comma– and colon–delimited list in any order (e.g., "1:4,6,9:12" or "6,1:4,9:12" or "1:4,9:12,6", etc). A colon–delimited interval is inclusive (e.g., "2:5" means rows 2, 3, 4, 5). If the number of rows specified does not equal the number of output files specified, the program exits with an error message. If no rows are specified, rows 1,2,...,n are written, where n is the number of output files in the outfile stack.

Example 1

```
dmtype2split in3.fits"[#row=1:3,6]" @outlist
```

From the type 2 input file, create 4 type 1 output files (1 each for rows 1, 2, 3, and 6). In this case the output file stack outlist must have 4 file names.

Example 2

```
dmtype2split in3.fits @outlist rows=""
```

From the type 2 input file, create type 1 output files (1 each for rows 1, 2, 3,..n where n is the number of output files in outlist).

Parameters

name	type	ftype	def	min	max	reqd	stacks
infile	file	input				yes	
outfile	file	output					yes
rows	string						
clobber	boolean		no				
verbose	integer		0	0	5		

Detailed Parameter Descriptions

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

The type 2 input file.

Parameter=outfile `(file filetype=output stacks=yes)`

The output dataset or stack.

Parameter=rows `(string)`

The rows specification specified in a comma− and colon−delimited list; any number of subintervals can be specified in any order.

As of CIAO3.0.0 this parameter is being deprecated. In future releases it will no longer be available. Please use a DM filter to select the desired rows.

Parameter=clobber `(boolean default=no)`

Specifies if an existing output file should be overwritten.

                                                                 Example 1

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

Specifies the level of verbosity in displaying diagnostic messages (0−5).

Bugs

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

See Also

dm

dmcols, dmfiltering, dmopt

tools

dmappend, dmarfadd, dmgroup, dmgti, dmjoin, dmmerge, dmpaste, dmsort, dmtcalc, tg_create_mask, tg_resolve_events, tgdetect, tgextract, tgidselectsrc, tgmatchsrc

---

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