URL: http://cxc.harvard.edu/ciao3.4/dmmerge.html
Last modified: December 2006

AHELP for CIAO 3.4

dmmerge

Context: tools

Jump to: Description Examples Parameters CHANGES IN CIAO 3.3 Bugs See Also

Synopsis

Merge two or more compatible tables (e.g. event files) into one.

Syntax

```
dmmerge  infile outfile [outBlock] [lookupTab] [columnList] [clobber]
[verbose]
```

Description

`dmmerge' takes a stack of tables which have the same columns and creates a single output table which concatenates the input tables, so that the number of rows in the output table is the sum of the number of rows in all the input tables. `dmmerge' is often used to merge several event files. Note that before merging event files, make sure that they have the same tangent point (see `ahelp reproject_events').

This tool supports the DM stack syntax ("ahelp stack"), meaning that the input files may be given as a comma−separated list or as a list in a text file. The default behavior is for dmmerge to combine all columns of the input event files. It is possible, however, to use the DM virtual file syntax ("ahelp dmsyntax") to select (or omit) certain columns or rows of the files. Rather than including the same list of columns on each input file, the columnList parameter applies a given column filter to each file in the input stack.

The intrinsic difficulty with this procedure is that `dmmerge' needs to combine several input headers into one header for the output file. A lookup table − specified in the lookupTab parameter − describes how header keys are to be merged.

Example 1

```
dmmerge "file1_evt2.fits,file2_evt2.fits" merged.fits outBlock=""
columnList=""
```

Merge all columns in the two event files into one output file.

Example 2

```
dmmerge "in1.fits[events],in2.fits[events],in3.fits[events]" out.fits
outBlock="" columnList="time,chipx,chipy"
```

Open three input files (specifically the 'events' block) and merge the data columns 'time,chipx,chipy'.

Example 3

```
dmmerge "@stack.lis[events][columns time,chipx,chipy]" out.fits
outBlock="" columnList=""
```

Open the 'events' block of the files listed in 'stack.lis' and extract the data columns 'time,chipx,chipy' for merging. This is the same as the previous example, but uses different syntax.

Example 4

```
dmmerge @stack.lis out.fits outBlock="" columnList=""
```

Open the files listed in the stack and use all data columns for merging.

Parameters

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

Detailed Parameter Descriptions

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

Input file name(s)

CIAO stacks may be a single filename, a comma−delimited list of names, or '@' followed by the name of a text file containing one filename on each line. Each filename may use the full CIAO DM virtual file syntax.

For one or multiple input files, there should be only one single output file.

Parameter=outfile `(file required filetype=output)`

Output file name

Parameter=outBlock `(string not required)`

Output block name

Output data block name. For a null string, it will use the same block name as the first input file, which is usually what you want.

dmmerge "evt.fits[events][cols time,pha][merged_events]" columnList="" outBlock="" is equivalent to

dmmerge "evt.fits[events]" columnList="time,pha" outBlock=merged_events

Parameter=lookupTab `(file not required)`

lookup table containing merging rules for header keys. The default value is a redirect to the system version, ${ASCDS_CALIB}/dmmerge_header_lookup.txt, which enforces CXC event pipeline header merging rules; see "plist dmmerge" for the full path of your system default.

Each line of the file contains a keyword and a merging rule. The rules include FORCE (always replace key), SKIP (always omit key), WarnOmit−tol (print a warning and omit if key valuess differ by more than tol), WarnFirst (print warning and use first value if keys are different), among others.

Parameter=columnList `(string not required)`

Column list

Apply a column filter to the input stack. "columnList=time,pha" is equivalent to appending "[cols time,pha]" to each entry in the input stack.

This parameter may also be used to omit certain columns. "columnList=−phas" is equivalent to including a "[cols −phas]" filter on the input files.

Parameter=clobber `(boolean not required default=no)`

Clobber output if it exists? [y/n]

Parameter=verbose `(integer not required default=0 min=0 max=5)`

Display informational messages

CHANGES IN CIAO 3.3

Parameter File

The outBlock, lookupTab, and columnList parameters have been changed to be hidden parameters. The kernel parameter has been removed, and the verbose parameter was added.

Bugs

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

See Also

chandra
: level

concept
: merging_rules

dm
: dmcols, dmfiltering, dmopt

tools
: celldetect, dmappend, dmarfadd, dmgroup, dmgti, dmimgcalc, dmjoin, dmpaste, dmsort, dmtcalc, dmtype2split, tgextract

The Chandra X−Ray Center (CXC) is operated for NASA by the Smithsonian Astrophysical Observatory.
60 Garden Street, Cambridge, MA 02138 USA.

URL:
http://cxc.harvard.edu/ciao3.4/dmmerge.html
Last modified: December 2006