

*AHELP for CIAO 3.4*

dmkeypar

Context: [tools](#)*Jump to:* [Description](#) [Examples](#) [Parameters](#) [CHANGES IN CIAO 3.4](#) [Bugs](#) [See Also](#)

Synopsis

Retrieve information about a keyword from an input file

Syntax

```
dmkeypar infile keyword [exist] [value] [rval] [ival] [sval] [bval]
[datatype] [unit] [comment] [echo] [mode]
```

Description

Searches the file header for the requested keyword. If the keyword exists, dmkeypar returns the value, datatype, unit, and comment associated with it. By default, this information is stored in the parameter file and may be accessed with any of the parameter tools, such as plist and pget. If "echo=yes", the value alone is also printed to the screen.

dmkeypar will exit with an error message if the user specifies an unreadable file, a non-existent keyword, or COMMENT or HISTORY as the keyword. In each of these cases, the "exist" parameter is set to "no" and all other parameters are reset to the default values. COMMENT and HISTORY are special keywords that can occur multiple times in a file, so it is not possible for dmkeypar to know which one specifically is being requested.

It is also possible to use this tool to get the value from a column in the FITS file for a single row. In this case, the "keyword" parameter is the name of the column, and the row must be specified in the "infile" parameter with the "[#row nn]" DM syntax (where nn is the row number). If a vector column is specified, only the first value in the pair is returned; to avoid problems, use the specific component, e.g. "y" instead of "sky". This functionality is illustrated in the "Examples" section.

Example 1

```
dmkeypar acis_evt2.fits mission
```

Retrieve the information for the keyword "mission" from the input file "acis_evt2.fits". Since the results are stored in the parameter file they can be accessed using the parameter tools. Here we use pget to list the the value and comment for the keyword:

```
unix% pget dmkeypar value
```

```
AXAF
unix% pget dmkeypar comment
Mission
```

The full information can be listed using pline:

```
unix% pline dmkeypar
infile='acisf01703N002_short.fits' keyword='mission' echo='no'
exist='yes' value='AXAF' rval='0' ival='0' sval='AXAF' bval='no'
datatype='string' unit='' comment='Mission' mode='ql'
```

Example 2

```
dmkeypar "acis_evt2.fits[3]" CCD_ID echo+
```

Retrieve the value of the keyword "CCD_ID" from the third block of the input file. This example uses the "echo=yes" option to print the results to the screen:

```
unix% dmkeypar "acis_evt2.fits[3]" CCD_ID echo=yes
7
```

Example 3

```
dmkeypar "indir/acisf01703N002_short.fits[#row=100]" EXPNO
```

dmkeypar is used to get the value from row 100 of the column EXPNO. The parameter file for dmkeypar can be examined – using pget and pline – to find the value:

```
unix% pget dmkeypar value
280
unix% pline dmkeypar
infile='indir/acisf01703N002_short.fits[#row=100]' keyword='EXPNO'
echo='no' exist='yes' value='280' rval='0' ival='280' sval=''
bval='no' datatype='integer' unit='' comment='' mode='ql'
```

Parameters

name	type	ftype	def	min	reqd
<u>infile</u>	file	input			yes
<u>keyword</u>	string				yes
<u>echo</u>	boolean		no		no
<u>exist</u>	boolean		no		no
<u>value</u>	string				no
<u>rval</u>	real				no
<u>ival</u>	integer				no
<u>sval</u>	string				no
<u>bval</u>	boolean		no		no
<u>datatype</u>	string		null	string real integer boolean null	no
<u>unit</u>	string				no
<u>comment</u>	string				no

Detailed Parameter Descriptions

Parameter=infile (file required filetype=input)

Input filename

Input FITS file with an optional extension in brackets, e.g. "filename.fits[GTI2]". By default, the tool reads the header from the principal block (e.g. EVENTS, EVENTS_IMAGE), not the primary block.

Parameter=keyword (string required)

Keyword to retrieve

COMMENT and HISTORY are invalid keywords, as explained in the DESCRIPTION. If using the tool to get information from a column and row, the column name is given in this parameter.

Parameter=echo (boolean not required default=no)

Echo value to screen?

Determines if the retrieved value is printed to the screen in addition to being stored in the parameter file.

Parameter=exist (boolean not required default=no)

Keyword existence

Set to "yes" if the keyword exists in the input file and "no" if it doesn't. Also set to "no" if the user attempts to search for a COMMENT or HISTORY keyword.

Parameter=value (string not required)

A pointer to the actual value of the keyword

This is a redirection pointer to the parameter in which the keyword value is stored, either rval, ival, sval, or bval.

Parameter=rval (real not required default=)

Keyword value -- real

If the value of the retrieved keyword is a real, it is stored in this parameter.

Parameter=ival (integer not required default=)

Keyword value -- integer

If the value of the retrieved keyword is an integer, it is stored in this parameter.

Parameter=sval (string not required)

Keyword value -- string

If the value of the retrieved keyword is a string, it is stored in this parameter.

Parameter=bval (boolean not required default=no)

Keyword value -- boolean

If the value of the retrieved keyword is a boolean, it is stored in this parameter.

Parameter=datatype (string not required default=null
min=string|real|integer|boolean|null)

Keyword datatype

The datatype of the specified keyword. Possibilities are "string", "real", "integer", "boolean", or "null".

Parameter=unit (string not required)

Keyword unit

If the keyword has a unit associated with it (e.g. seconds, pixels), the information is stored in this parameter.

Parameter=comment (string not required)

Keyword comment

The comment associated with a keyword, if present.

CHANGES IN CIAO 3.4

IRAF QPOE Files

Support for IRAF QPOE files has been removed from CIAO.

Bugs

See the [bugs page for this tool](#) on the CIAO website for an up-to-date listing of known bugs.

See Also

concept

[parameter](#)

modules

[paramio](#)

paramio

[paccess](#), [paramclose](#), [paramopen](#), [pget](#), [pgets](#), [plist_names](#), [pquery](#), [pset](#), [punlearn](#)

tools

[dmdiff](#), [dmhedit](#), [dmhistory](#), [dmmakepar](#), [dmreadpar](#), [paccess](#), [pdump](#), [pget](#), [pline](#), [plist](#), [pquery](#), [pset](#), [punlearn](#)

Ahelp: dmkeypar – CIAO 3.4

Smithsonian Institution, Copyright © 1998–2006. All rights reserved.

