



Bugs: dmgti

A list of bugs fixed in CIAO 3.4 is included at the end of this document.

Bugs

1. *When using status bits in an expression, only the last condition is checked.*

When using status bits in an expression such "status==X1F,X3T,X18F", only the last condition (here, "X18F") is checked.

Workaround:

Set the logic explicitly:

- ◆ logical "AND": status=((bits==X1T)&&(bits==X2T))
- ◆ logical "OR": status=((bits==X1T)|| (bits==X2T))

2. *Intersecting a new GTI with an existing file does not work. (01 Dec 2006)*

If the name of an existing GTI file is given as the outfile parameter of another dmgti run, the tool is supposed to modify it in place to intersect the new GTI with the old GTIs. This bug results in the output file containing only the GTI for the last userlimit condition.

Workaround:

Run dmgti multiple times to create separate GTI files, then use dmcopy to apply the GTI filters to the data.

Bugs fixed in CIAO 3.4

The following is a list of bugs that were fixed in the CIAO 3.4 software release.

1. *Use of time information by dmgti causes filter to appear shifted (generally seen for lightcurves)*

dmgti uses the TIMEPIXR and TIMEDEL keywords to modify the TIME column when creating GTI files. TIMEPIXR defines whether or not the TIME column should be taken to mean the beginning (TIMEPIXR=0), the middle (TIMEPIXR=0.5) or end (TIMEPIXR=1) of the bin. TIMEDEL records the time resolution of the data; this is the bin size between rows for a binned dataset or the resolution of the time stamp for event lists. Refer to "ahelp_chandra_times" for further details on the time information in Chandra data.

Bugs: dmgti – CIAO 3.4

The way that `dmgti` uses the time information may result in a filter that appears shifted from the expected times for files where `TIMEPIXR` is not equal to 0. This is generally seen when creating filters for lightcurves.

Workaround:

To get the desired filter when working with lightcurves, rename the `TIME_MIN` column in the lightcurve input to `dmgti` to `TIME`. This means that the GTI filter will actually be determined using the `TIME_MIN` values.

Assume that the original lightcurve created by `dmextract` is called "lc.fits". This `dmcopy` command

1. copies the `TIME` column to `TIME_ORIG`
2. copies the `TIME_MIN` column to `TIME`
3. copies the rest of the columns ("*") to the output file as-is

```
unix% dmcopy "lc.fits[cols TIME_ORIG=TIME,TIME=TIME_MIN,*]" lc_cols_new.fits
```

Then use the new file in the `dmgti` command, e.g.

```
unix% dmgti lc_cols_new.fits lc_rate_0.06.gti userlimit="count_rate<0.06"
```

For more information on working with lightcurves, refer to the [CIAO Timing Threads](#).

2. ***The tool does not check that the input virtual file syntax is valid.*** (13 Jun 2006)

It will segmentation fault if the input filter uses a column name that does not exist in the file.