



Bugs: ChIPS

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

Caveats

1. *Certain characters should be avoided in directory names and filenames as they can cause problems in I/O.*

- ◆ Often problems can be avoided by using double quotes when using "special" characters.
- ◆ The standard set (defined as special by most operating systems) of "special" characters is:

```
& ( ) [ ] ; | * ? ' " ` $ < > ; { } ^ # / \ (space character)
```

Sherpa and ChIPS have additional restrictions on the use of the following:

```
: +
```

- ◆ Sherpa and ChIPS are known to crash when trying to write files whose names contain:

```
( ) & |
```

- ◆ Sherpa and ChIPS will not function properly if the directory path contains:

```
: ( )
```

2. *If contour plots appear jagged when plotting with log scales, the value of `chips.mingridsize` should be increased.*

To get ChIPS to recalculate the contours, switch from log to linear scales and then back again.

Bugs

General

1. *The ChIPS window refreshes inconsistently under heavy use.*

Workarounds:

1. If you are making many modifications to the plot and do not need to see the changes until all modifications are done, then call `REDRAW OFF` before making the changes and then `REDRAW ON` once you have made the changes. The functions `chips_redraw()` and `chips_auto_redraw()` can be used from S-Lang scripts.
 2. If you are creating a postscript plot and do not need the X-windows display, call `BATCH ON` before creating the plot.
2. *To include underscores in plot titles, they must be escaped (i.e. `_` instead of `_`); this is due to a conflict with the LaTeX emulation.*

Plotting

1. The `chips_get_*range()` functions return different values depending on whether the axis is being drawn in log or linear space.

This is contrary to how the `chips_set_*range()` commands work.

2. ChIPS converts double-precision numbers to floating-point values before plotting them, which can cause problems for values that are either too large or too small.

The allowed range is approximately $1e-38$ to $3e38$ (for both positive and negative values).

Workaround:

If possible, normalize the data – e.g. by subtraction or division – before plotting so that it can be represented by floating-point numbers. For example, instead of

```
chips> curve "evt2.fits[cols time,energy]"
```

try

```
chips> clear
chips> d = readfile("evt2.fits[cols time,energy]")
chips> () = curve( d.time - d.time[0], d.energy )
```

Printing

1. The output of `print postfile` may be clipped even if the `pagesize` is set correctly; this is most likely to affect plots oriented as "landscape" layout.

The postscript drivers work well for plots no bigger than 8x8 inch or portrait plots up to about 8x10 inches. Any other setup causes problems due to inconsistent handling of margins that causes errors in calculating the bounding box.

Workarounds:

1. Edit the bounding box by hand in PS file
2. Delete the bounding box definition from the PS file
3. Use "portrait" layout or a smaller size output for your plots
4. gv has the option to change or ignore the bounding box when viewing a PS file

Saving & Restoring Plots

1. If the command "store file" is issued in ChIPS and `file.fits` already exists, the data gets appended to `file.fits` rather than overwriting the file.

This can lead to plots that don't work or look strange when the plot is restored.

Workaround:

Use a unique filename or manually remove the existing file before writing the new data.

2. When a save file is created, it stores explicit limit values that are used during the restore; these values differ slightly from the limits that are drawn in the original plot.

Workarounds:

1. Explicitly specify a limits command before doing the store command – this will ensure that the same limits are used for both the original and the restored plot.
2. Let ChIPS chips automatically calculate the limits in the restore plot. After performing the restore, try typing:

```
chips> limits auto auto
```

The plot should shift limits to match the original.

3. **ChIPS will not correctly restore a state file if the `drawarea` command is commented out or removed.**

The message "You do not have a Current Drawing Area." is printed to the screen in this case.

S–Lang–related

1. **ChIPS does not emit errors for illegal S–Lang statements**

For example:

```
chips> () = evalfile()  
chips> () = evalfile(dsfhksdlfhdslkfs  
chips> () = evalfile(dsfhksdlfhdslkfs  
chips> () = evalfile("bob")
```

Bugs fixed in CIAO 3.4

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

1. ***libgcc_s.so.1 must be installed for pthread_cancel to work***

The `libgcc_s.so.1` file is packaged with CIAO for users who do not have it installed on their system. If the library is already installed on the system, a harmless warning is printed when exiting *ChIPS*:

```
chips> exit  
libgcc_s.so.1 must be installed for pthread_cancel to work
```

To avoid this warning, rename the CIAO copy of the library (`$ASCDS_INSTALL/lib/libgcc_s.so.1`) and the version available on the system will be used in its place.

This file will be packaged with CIAO 3.4, but will be renamed so that it is not picked up by default. Users who don't have the file can rename it for use with CIAO; refer to the [Linux8 package note](#) in the [Installing CIAO thread](#).

Bugs: ChIPS – CIAO 3.4