




---

 AHELP for CIAO 3.4

# gui

Context: gui

*Jump to:* [Description](#) [CHANGES IN CIAO 3.0](#) [See Also](#)

---

## Synopsis

Graphical User Interface (GUI)

## Description

A Graphical User Interface (GUI; pronounced "gooey") is an alternative to command-line tool use. The purpose of a GUI is to give users an intuitive, visual interface for interacting with their data. CIAO contains a number of GUIs, as listed below. This file describes those topics relevant to all the GUIs; for help on an individual gui use "ahelp <gui name>".

GUI	Brief description
firstlook	Provides a quick means of viewing the contents of Chandra data and performing some simple tasks – view images, extract spectra, and create lightcurves – taking advantage of ds9, prism, and filtwin.
peg	peg is a graphical parameter editor. It allows you to set parameters for tools and run them without having to use command-line tools like pset and plist.
prism	Provides the user with the ability to view (both numerically and graphically) the contents of data files.
filtwin	Provides graphical access to the powerful filtering and binning capabilities afforded by the Data Model (dm) library.
taskmonitor	GUI to run a task as a background process and display the output.

## Command-line options

Apart from toolagent, all the GUIs support the "--help" command-line option which lists the available options, as shown in this example with prism:

```

unix% prism -help

prism, version CIAO3.0
Copyright (C) 2003 Smithsonian Astrophysical Observatory

Usage: prism [FILE] [options]

Options:
```

```
-help | -h | -H      : display available options
-toolmenu <file>    : load the specified tool menu file
-xpa <name>         : set communication access point to name
-v | -V             : version info
```

The `-toolmenu` option is described in "ahelp analysis-menu". The `-xpa` option allows you to change the name of the XPA access point of the tool. For firstlook you need to use `-info` to get the options, which are different to those above.

Even though not listed in the help messages of the tools, all the GUIs – including firstlook – understand the `-iconic` option which suggests that the GUI should start up in an "iconified" or hidden state (note that window managers may ignore this request). Similarly, all GUIs except firstlook have a `-nosession` option that indicates that this instance of the GUI should not be considered to be part of a CIAO session (see the "Sessions" section below).

## Running CIAO Tools from a GUI

All the GUIs except for firstlook contain a user-defineable Analysis Menu from which tools can be launched. Whilst the CIAO distribution comes with a default setting for the menu contents – see the text file `$ASCDS_INSTALL/bin/ciao.ans` – it is easy to configure a menu tailored to your own requirements. The menu format uses the same system as ds9, which means you can use the same menu in ds9 and the CIAO GUIs. See "ahelp analysis-menu" for more details.

## Sessions

While the GUI tools can be run individually, they can also be viewed as an integrated analysis environment. In CIAO this is termed a "session", and provides – amongst other things – a means for saving/restoring the state of a particular analysis; the firstlook tool makes use of the session concept to allow easy manipulation of the dataset in different GUIs. See "ahelp session" and "ahelp ciao.par" for more information.

## Data Display

CIAO uses the SM and ds9 packages to provide plotting and imaging capabilities; both are provided as part of the CIAO installation (SM is used internally by ChIPS and is not available directly).

## Configuring the look of the GUIs

The CIAO software may utilize a resource file to perform user dependent configuration of the GUIs. The configuration file contains entries for items such as imager preference (ds9 vs saotng), dynamic vs private color maps, application foreground and background colors, and fonts.

Since the resource file contains many X/motif related entries, it is treated in a similar manner to X resource files. A hierarchy exists regarding from where this file is utilized:

- First, a user's home directory is checked for the existence of a '.CXCdefaults' file. If a readable copy of the file is found, it is used.
- If the file is not found in the user's home directory, the code then checks whether the user has set their XENVIRONMENT environment variable to specify where to obtain the .CXCdefaults file – both the path and filename must be specified, e.g. "setenv XENVIRONMENT /pool14/my\_stuff/.CXCdefaults".

- If a .CXCdefaults file is not found in either of the previous two locations, the \$ASCDS\_INSTALL/config/system directory is checked for a .CXCdefaults file (\$ASCDS\_INSTALL is automatically set to the root of the CIAO installation).
- Finally, the \$ASCDS\_INSTALL/config/system directory is checked for 'CXCdefaults' (without the preceding '.'). In standard CIAO installations, this location contains a copy of the file. If the file does not exist here for any reason, internal defaults are used.

Depending on how a user configured the CIAO environment, there may or may not be a local copy of the .CXCdefaults file in the home directory. One is of course always free to copy the file into the home directory and make customizations.

## Setting the colors used by the GUIs

The GUIs can install private colormaps if configured to do so. This can be used to ensure that the GUI starts with the correct color scheme or to avoid the "flashing" that can occur when moving the mouse into/out of the GUI window. To change the default behavior (listed below) requires changing the CXCdefaults resource file (described above).

The resource value `"*forcePrivateColormap"` in the CXCdefaults resource file controls whether or not the GUIs use a private colormap. The value may be set to one of three values:

- True – X gui applications always install their own colormaps
- False – X gui applications always use the default colormap
- dynamic – X gui applications only use a private colormap if necessary

The default value of this resource is set to dynamic. Since the use of a private colormap will result in color flashing (the color maps will swap in/out so that the window with pointer focus will have the correct color scheme), the applications will use the default colormap unless they are unable to allocate their own colors. Color intensive applications such as netscape and the CDE window manager often consume the available cells of the default colormap forcing the CIAO GUIs to either switch to a private colormap or not be able to display items in the desired colors. If a user does not want to experience color flashing, he should set the resource `*forcePrivateColormap` to False (note that this may cause some data to be unreadable if the background and text get set to similar colors).

Also note that the color scheme for the X based GUI applications is controlled by the following CXCdefault resources: `<GUI name>*background` and `<GUI name>*foreground`, where `<GUI name>` is one of `filtwin`, `peg`, or `prism`. If the values exist in the user's CXCdefaults file, they will be used to setup the user's color scheme (adhering to the colormap specifications outlined above). If they do not exist, the colors of the applications will generally be the colors specified by the `'*background'` and `'*foreground'` resources, but this is window manager dependent. For instance, CDE will override these values and use its own background color.

## Changes to the gridtype parameter

If you see the following message when running a CIAO GUI:

```
Warning: Value for GridType is deprecated and will be removed ...
```

then you need to either update your personal copy of the CXCdefaults resource file so that the line

```
*MatrixTable.gridType : XmGRID_LINE
```

is changed to

```
*MatrixTable.gridType : XmGRID_CELL_LINE
```

or delete the file.

## CHANGES IN CIAO 3.0

### Addition of peg

The parameter editor GUI (peg) was added to CIAO. See "ahelp peg" for more information.

### Analysis menu

The file format used to describe the contents of the Analysis Menu in the GUIs was changed to match that used by ds9. See "ahelp analysis-menu" for more information.

### The `-iconic` command-line option

A new command-line option, "`-iconic`", has been added for the GUIs.

## See Also

*calibration*

[caldb](#)

*chandra*

[coords](#), [guide](#), [isis](#), [level](#), [pileup](#), [times](#)

*chips*

[chips](#)

*concept*

[autoname](#), [configure](#), [parameter](#), [session](#), [stack](#), [subspace](#)

*dm*

[dm](#), [dm binning](#), [dmcols](#), [dmfiltering](#), [dmimages](#), [dmimfiltering](#), [dmintro](#), [dmopt](#), [dmregions](#), [dmsyntax](#)

*gui*

[analysis-menu](#), [ciao.par](#), [ciaoshmem](#), [filtwin](#), [firstlook](#), [peg](#), [prism](#), [taskmonitor](#)

*modules*

[paramio](#), [pixlib](#), [stackio](#), [varmm](#)

*slang*

[overview](#), [slang](#), [tips](#)

*tools*

[mkoif](#)



