Chandra X-ray Center

AHELP for CIAO 3.4

pset

Context: paramio

Jump to: Description Examples See Also

Synopsis

Set a parameter value within S–Lang.

Syntax

```
pset( paramfile, parname, value )
pset( paramfile, pars_assoc_array )
```

Description

This function is essentially the same as the command–line version (see "ahelp tools pset"). To use pset() the paramio module must have been loaded.

As with most of the paramio routines, the parameter file can be specified by either name or using the output of paramopen(). To set just one parameter value you use the three–argument form of pset(), namely:

```
pset( paramfile, parname, value )
```

To set the values of multiple parameters you use the two–argument form where the second argument is an associative array, where the keys are the parameter names and their values are the new parameter values.

Error handling

As with all the paramio routines, the PF_Errno variable is set to 0 on success, or on error it is set to one of the error codes listed in the paramio documentation.

Example 1

```
chips> require("paramio")
chips> punlearn("dmcopy")
chips> pset("dmcopy","infile","in.fits")
chips> pget("dmcopy","infile")
in.fits
chips> pset("dmcopy","verbose","3")
chips> pget("dmcopy","verbose")
3
```

Here we use the pset() function to set the "infile" and "verbose" parameters of "dmcopy". We then use pget() to check the values.

Example 2

```
chips> punlearn("dmcopy")
chips> pars = Assoc_Type []
chips> pars["infile"] = "in.fits"
chips> pars["outfile"] = "out.fits"
chips> pars["clobber"] = "yes"
chips> pset( "dmcopy", pars )
```

The pset() command can set multiple parameters for a tool at one go by using an associative array. The keys of the array are the parameter names, and the stored values are used to set the parameter values.

To see how the above call to pset() has changed the parameter file we list the parameter file contents using the command–line tool plist:

```
chips> !plist dmcopy

Parameters for /home/ciaouser/cxcds_param/dmcopy.par

      infile = in.fits          Input dataset/block specification
     outfile = out.fits         Output dataset name
    (kernel = default)          Output file format type
    (option = )                 Option – force output type
   (verbose = 0)                Debug Level
   (clobber = yes)              Clobber existing file
      (mode = ql)
```

Example 3

```
chips> punlearn("dmcopy")
chips> fp = paramopen("dmcopy")
chips> pset(fp,"verbose","3")
chips> pget(fp,"verbose")
3
chips> paramclose(fp)
chips> pget("dmcopy","verbose")
0
```

If called with the name of the parameter file, then pset() opens the file with a mode of "rwL" which means that the value will be stored in the file even if the parameter is a hidden one. This behaviour is shown in the first example when pset() is used to set the value of the "verbose" parameter.

However, in the current example we have used paramopen() to open the parameter file, and have done so with the mode set to "r" (the default value). Therefore the "verbose" parameter will only equal "3" whilst the parameter file is still open. Once it is closed all knowledge of changes to the parameter file are lost, which is why the final pget() call returns a 0 instead of 3.

If the paramopen() call is changed to 'fp = paramopen("dmcopy","rwL")' then the final pget() call would return 3.

See Also

concept
> parameter

Example 2

modules

      paramio

paramio

      paccess, paramclose, paramopen, pget, pgets, plist_names, pquery, punlearn

tools

      dmhistory, dmkeypar, dmmakepar, dmreadpar, paccess, pdump, pget, pline, plist, pquery, pset, punlearn

---

The Chandra X−Ray Center (CXC) is operated for NASA by the Smithsonian Astrophysical Observatory.
60 Garden Street, Cambridge, MA 02138 USA.
Smithsonian Institution, Copyright © 1998−2006. All rights reserved.

Example 2