IDL_HS Command Line & Scripting

James Tappin

26th June 2002

Contents

1 Introduction

1.1 History

Historically `IDL_HS` began life as a command-line tool for the simple reason that at the time that it was first conceived, IDL did not have widgets and many users did not have access to workstations that supported windows and widgets anyway. After a few years, the command-line interface became sufficiently large that nobody could actually remember all the settings without a crib-sheet of some kind. By this time support for widgets was becoming much more widespread and so the familiar version 3 widget interface was developed. The version 3 interface was in effect an additional layer which wrapped the command-line procedures—effectively acting as a crib-sheet.

With the advent of version 4 the widget interface is integrated into the object-oriented structure of the environment. In principle it would be possible to do all the necessary operations that might need to be done from the command-line (or a user program) using the low-level object methods, however few users are familiar with IDL objects and the syntax is not really convenient for interactive use or for quick scripting. Because of this, a basic command-line interface/scripting system has been written for `IDL_HS` version 4; this is essentially a set of wrappers for the various setting methods.

1.2 Basics

Unlike the version 3 command-line routines, those in version 4 do not support interactive operation, however they all have a `HELP` keyword which runs `DOC_LIBRARY` (or `XDL` if widgets are present) for the routine.

Generally speaking, when a routine is not applicable, it will generate an error message and return without doing anything, for example if you were to call a PAD dataset specific routine when the current dataset is a rate dataset..

In the listings that follow, the routines and their purpose are listed; to get more details either use the reference manual or use the `help` keyword on the routine itself. Almost all of the routines in the command line interface are procedures rather than functions. The exceptions to this are generally routines that allow access to the object-oriented core of the environment (e.g. `get_current_ds` which returns the object reference of the current dataset) and they are generally intended to be used by the procedures. In the lists that follow, functions are given in italics.

2 Top Level

The top-level CLI routines are those that operate on the overall environment. Technically, they interface to the hs_container methods of the core.

browse_hsio Start up a browser to examine the contents of an HSIO file.

current_dataset Select a new current dataset. The current dataset is the one on which all dataset operations are performed.

delete_dataset Delete a dataset from the environment.

get_container *Return the object reference of the master container object*

get_current_ds *Return the object reference of the current dataset.*

get_plot_obj *Return the object reference of the plot-device control object.*

hs_help Start the help system.

hs_menu Start up the widget interface.

list_datasets Generate a quick list of the datasets present in the environment.

new_dataset Add a new dataset to the environment.

new_environment Delete the current environment and create a new one.

replace_environment Replace the current environment with a previously saved one.

save_all Save the whole environment to an IDL save file.

3 IDF routines

The IDS routines allow you to change the settings of the various Instrument Definition Files which are used to specify energy ranges, geometry factors etc.

lan_idf_info Get information from the LAN rates IDF file.

lan_idf_set Set the IDF file for "ordinary" LAN rates

list_path Get the data search path for an IDF definition.

mfsa_idf_info Get information from the MFSA IDF files.

mfsa_idf_set Set the IDF file for MFSA data

remove_path Remove a directory from a search path.

set_path Set the search paths for the various data classes.

trk_idf_info Get information from the track IDF data.

trk_idf_set Set the track map version for Track and PHA matrix data.

uds_idf_add_spec Add a new UDS data type.

uds_idf_delete_instrument Remove an instrument fromt eh list of defined UDS instruments.

uds_idf_info Get information from the UDS IDF files.

uds_idf_update Rescan for new SPEC files.

uleis_idf_info Get information from the ULEIS IDF files.

4 Dataset routines

All the dataset routines operate on the current dataset. Not all routines are applicable to all type of dataset. In general if you attempt to use a routine on an inappropriate type of dataset or without a current dataset selected, then a message will be generated and nothing will be done. The `HELP` output includes the types of dataset to which the routine is applicable.

act_class Set flare classes for activity plots

act_cmp Set CMP plotting for activity plots

act_get_data Get the data from an activity plot.

act_header Control plotting of headers on activity plots.

act_histogram Set up histograms on activity plots

act_no_optical Control plotting of unlocated flares in activity plots

act_scale Set symbol size for activity plots.

act_system Set rotation system for activity plots.

add_plot Add a dataset-specfic plot control.

add_stream Add a stream to the current dataset.

average Set the averaging interval for a dataset

background Set whether to subtract backgrounds

colour_table Select colour table for matrix and dynamic spectrum datasets

dataset_info Return information about the current dataset.

delete_panel Delete a panel from the current dataset

delete_stream Delete a stream from the current dataset.

display Display the dataset.

do_edit Apply an edit to the current dataset.

dyn_spect_layout Set layout for dynamic spectrum datasets

dyn_spect_scaling Set the scaling for a dynamic spectrum dataset

editor Start the data editor. This needs widgets to be available.

errors Set whether to display error bars

flux Set whether to plot fluxes of count rates

get_ds_plot_obj Return the plot control object for the dataset.

list_stream_types List the type of stream supported by the current dataset.

matrix_get_data Return the data from a matrix dataset.

matrix_header Set whether to add a plot header for matrix datasets

matrix_hist_delete Delete a histogram from a matrix dataset.

matrix_hist_display Display a histogram of a matrix dataset.

matrix_histogram Create a new histogram in a matrix dataset.

matrix_locate Mark positions and give information about them.

matrix_maps Select maps to display on matrix datasets

matrix_scale Set scaling options for matrix datasets

matrix_sum Set summing option for matrix datasets

matrix_view Set the view region of a matrix dataset

move_panel Move a panel in the current dataset.

move_stream Move a stream in the current dataset

pad_axis Select axis for PAD datasets

pad_get_data Return the data from a PAD dataset.

pad_hide_last Control last-sector hiding for PAD datasets.

pad_join Set joining option for pad datasets

pad_layout Set layout for pad datasets

pad_look_system Set coordinates for look plots

pad_spacing Set spacing between plots in PAD dataset

rate_get_data Return the data from a rate dataset.

rate_homogeneity Set the level of mismatch to tolerate in overplotted streams.

rate_layout Set layout for rate datasets

rate_offsets Set trace offsets in rate datasets

rate_scaling Set the scaling for a rate dataset

save_dataset Save the dataset to an IDL save file.

sectors Set the default sectoring for a dataset

set_archive Select the archive type for LAN rates data

set_energy_unit Select whether to work in MeV or keV.

set_name Set the name of the dataset

set_time Set the time range of the dataset

show_details Show details of the current dataset

spect_get_data Return the data from a spect dataset.

spect_join Set the joining option of a spect dataset

spect_layout Set the layout of a spect dataset.

spect_scaling Set the scaling of a spect dataset

write_dataset Write the dataset to an ascii file.

x_level Set the highest P channel from which to exclude the Sun sectors.

5 Plot control

These routines set various options for the plot device. The keywords control whether this is the global or the dataset control set.

locate Mark positions and display information about them.

plot_char_size Set the character size to use.

plot_close Close (and spool) a plot.

plot_ctable Set the colours for the various traces

plot_info Return various information about the plot state.

plot_open Open the plotting system.

plot_ps_opts Set various PostScript options

plot_ps_path Set the directory in which to put the PostScript files

plot_ps_spool Set the command(s) for spooling PostScript files

plot_set Set the plot device.

plot_size Set the size of the page or window for the plot

plot_stack Add the current plot transforms to the stack.

plot_thick Set the thickness of the lines to use.

plot_x_opts Set various X-window system options.

plot_zone Set the part of the page or window to use.

6 Streams

The stream routines act on individual streams of the current dataset. The stream is specified by a 1-based stream index. For spect and PAD datasets the index is a 2-element array giving the panel number and the the position of the stream in the panel.

Not all types of stream support all the operations; neither Matrix nor activity plot datasets support stream operations as they have but a single stream.

dyn_spect_stream_data Return the data from a single stream of a dynamic spectrum dataset.

form_indices Calculate spectral index streams

get_stream *Return the reference of the selected stream.*

pad_stream_data Return the data from a single stream of a PAD dataset

rate_calc Compute derived rate streams.

rate_stream_data Return the data from a single stream of a rate dataset

sector_diffs Compute sector difference streams.

sector_ratios Compute sector ratio streams.

spect_calc Compute derived spectral streams

spect_stream_data Return the data from a single stream of a spect dataset.

stream_background Set the background copunt rates for a stream

stream_hide Hide/unhide a stream.

stream_info Return various items of information about a stream.

stream_name Set a mnemonic name for a stream

stream_sectors Set the sectoring for a stream

stream_yscale Set the y-axis scaling for a stream

7 Markers

These routines allow the manipulation of markers.

marker_add Add a marker to the current dataset

marker_delete Delete a marker from the current dataset

marker_hide Hide or show a marker in the current dataset

marker_set Modify the properties of a marker.

8 Other

In addition to the command-line interface outlined above, the `rc_maker` and `uds_wizard` configuration tools are designed to be run from the `IDL>` prompt.

9 An example

This is a short example code to show the use of the command line interface. It makes a rate dataset that has the “de1” electron channels from both HISCALE and EPAM. It displays the and then makes the data arrays available to the main program.

```
; This is a demonstration of the scripting capabilities of
; IDL_HS version 4.3

new_dataset, /rate, name='Scripting demo'
set_time, [99,100],[0,250]
add_stream, 'hiscale', 'de1'
add_stream, 'epam', 'de1'
```

```
set_archive, /epam,/uaf
set_archive, /hiscale,/cuaf
average, 6, /hours

display

rate_get_data,rates,times

help, rates, times

end
```