
Subject: printing widget, as-is
Posted by [Eric E. Dors](#) on Thu, 03 Sep 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

<!-- This file was generated by mk_html_help.pro -->
<html>

<head>
<TITLE>Extended IDL Help</TITLE>
</head>

<body>
<H1>Extended IDL Help</H1>
<P>
This page was created by the IDL library routine
<CODE>mk_html_help</CODE>. For more information on
this routine, refer to the IDL Online Help Navigator
or type: <P>
<PRE> ? mk_html_help</PRE><P>
at the IDL command line prompt.<P>
Last modified: Tue Jan 27 22:56:55 1998.<P>

<HR>

<H1>List of Routines</H1>

EXPAND_ALL_VARS
FORM_PRINT_CMD
GET_CFG_LINE
GET_CONFIG_FILENAME
PRINT_FINISH
PRINT_WINDOW
READ_PRINT_CONFIG
<P>

<HR>

<H1>Routine Descriptions</H1>

<H2>EXPAND_ALL_VARS</H2>
[Next Routine]
[List of Routines]
<PRE>
NAME:
 expand_all_vars

PURPOSE:

This function takes a list of variable definitions and returns the expanded list. That is it expands references of variables on the right hand side to their definitions

CATEGORY:

General Utility

CALLING SEQUENCE:

Result = expand_all_vars(varlist, expanded)

INPUTS:

varlist: A 2xn dimensional array of variable and definitions
[[name1,def1],[name2,def2],...,[namen,defn]]

KEYWORD PARAMETERS:

top_widget: the id of the parent widget base if it exists

OUTPUTS:

result: 0 for failure, 1 for success.

expanded: The two dimensional variable list with variable references in the definitions expanded using other variables in the list.

PROCEDURE:

The variable list is looped through up to max_iter times. Each time through it looks at each variable definition one by one. It then looks for the first occurrence of a variable reference in the definition, and then replaces it with the appropriate definition. Then it moves on to the next variable and keeps looping until all the variables references in the definitions have been replaced, an undefined variable is found, or the max_iter has been hit.

EXAMPLE:

result = expand_all_vars(varlist,expanded)

Written by: Eric E. Dors, 9 September 1997.

MODIFICATION HISTORY:

Fri Jan 23 19:42:12 1998, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Added top_widget parameter for use with widget_message.

</PRE><P>
(See print_window.pro)<P>
<HR>

<H2>FORM_PRINT_CMD</H2>
[Previous Routine]
[Next Routine]
[List of Routines]
<PRE>
NAME:
 form_print_cmd

PURPOSE:

The purpose of this function is to create a string which contains the operating system command to print a file of a given type to the correct printer.

CATEGORY:
 General Utility

CALLING SEQUENCE:

```
Result = form_print_cmd(print_file, printer_type, $  
                        print_color, lpr_switches, ptype_btns, $  
                        color_btns, print_command)
```

INPUTS:

print_file: The name of the file to be printed.

printer_type: A value from the list of printer types in ptype_btns describing the type of printer.

print_color: A value from the list of printer color support in color_btns describing the color printing options selected.

lpr_switches: A string of user printer switches to be used in addition to the switches set in print_window.cfg.

ptype_btns: A list of printer types.

color_btns: A list of printer color options.

KEYWORD PARAMETERS:

top_widget: the id of the parent widget base if it exists

OUTPUTS:

print_command: A string which contains the system command necessary to print the desired file.

result: 0 for failure of this routine, 1 for success.

PROCEDURE:

Replace the reserved variable queue with the print queue appropriate to the current print job. Replace the reserved variable file with the file name appropriate for this print job. Expand all other variables. Return the print_command requested.

EXAMPLE:

```
ret_val=form_print_cmd(print_file, printer_type, $
    print_color, lpr_switches, ptype_btns, $
    color_btns, print_command)
```

```
if (ret_val eq 1) then begin
    spawn,print_command
endif
```

Written by: Eric E. Dors, 9 September 1997.

MODIFICATION HISTORY:

Fri Jan 23 19:42:25 1998, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Added top_widget parameter for use with widget_message.

Fri Sep 12 15:36:07 1997, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Add recognition of new variable to configuration file called 'file' this shows the position of the file name in the print command.

</PRE><P>
(See print_window.pro)<P>
<HR>

<H2>GET_CFG_LINE</H2>
[Previous Routine]
[Next Routine]
[List of Routines]

<PRE>

NAME:

get_cfg_line

PURPOSE:

The purpose of this function is to return one variable
expression line configuration file.

CATEGORY:

General Utility

CALLING SEQUENCE:

retval = get_cfg_line(unit, oneline)

INPUTS:

unit: The file unit number of the opened configuration file.

OUTPUTS:

oneline: One variable expression is returned from the
configuration file. Comment lines are skipped.

retval: 1 is returned on success, 0 is returned if eof is
found before a variable definition is found.

PROCEDURE:

The configuration file is read in line by line until a non
commented line is found and then that line of text is returned
to the caller in oneline and 1 is returned in retval. If eof
is hit then the value of oneline is not guaranteed to be
anything useful, and retval is set to 0.

EXAMPLE:

retval = get_cfg_line(unit, oneline)

Written by: Eric E. Dors, 9 September 1997.

MODIFICATION HISTORY:

```
</PRE><P>
<STRONG>(See print_window.pro)</STRONG><P>
<HR>

<A NAME="GET_CONFIG_FILENAME">
<H2>GET_CONFIG_FILENAME</H2></A>
<A HREF="#GET_CFG_LINE">[Previous Routine]</A>
<A HREF="#PRINT_FINISH">[Next Routine]</A>
<A HREF="#ROUTINELIST">[List of Routines]</A>
<PRE>
```

NAME:

get_config_filename

PURPOSE:

This function locates the fully qualified filename of the configuration file required for the operation of print_window.pro.

CATEGORY:

General Utility

CALLING SEQUENCE:

Result = get_config_filename(filenameout)

KEYWORD PARAMETERS:

top_widget: the id of the parent widget base if it exists

OUTPUTS:

result: result is set to 1 for success and to 0 for failure of this subroutine to find the configuration filename.

filenameout: The fully qualified filename of the configuration file print_window.cfg.

PROCEDURE:

First an optional environment variable called PRINT_WINDOW_CFG is searched for. This environment variable is supposed to contain the path to the configuration file "print_window.cfg". If the environment variable is not found then directory in

which print_window.pro resides is searched for the system default configuration file also called print_window.cfg. If the file is found the configuration information is read in, otherwise the programs writes an error message and exits.

EXAMPLE:

```
F = get_config_filename(cfgfilename)
```

Written by: Eric E. Dors, 9 September 1997.

MODIFICATION HISTORY:

Fri Jan 23 19:41:41 1998, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Added top_widget parameter for use with widget_message.

```
</PRE><P>  
<STRONG>(See print_window.pro)</STRONG><P>  
<HR>
```

```
<A NAME="PRINT_FINISH">  
<H2>PRINT_FINISH</H2></A>  
<A HREF="#GET_CONFIG_FILENAME">[Previous Routine]</A>  
<A HREF="#PRINT_WINDOW">[Next Routine]</A>  
<A HREF="#ROUTINELIST">[List of Routines]</A>  
<PRE>  
NAME:  
    PRINT_FINISH
```

PURPOSE:

This procedure completes the actions initiated by PRINT_WINDOW.

CATEGORY:

General Utility

CALLING SEQUENCE:

PRINT_FINISH, handle

KEYWORD PARAMETERS

parent_id: The widget handle to the controlling base widget.
This is used so that informational dialog can be centered on that widget.

noconfirm: if 0 then ask before deleting print file,
otherwise delete without confirmation.

COMMON BLOCKS:

print_event: A common block to pass information between the event handler and the widget creation routine.

print_finish: A common block to hold the selections made during the PRINT_WINDOW dialogue with the user.

SIDE EFFECTS:

The current plot device is changed.

If it was requested either a file is created, printed or both.

EXAMPLE:

```
result=print_window(xsize=draw_x, ysize=draw_y)
...
if (result) then begin
  data plotting routines go here
  time_stamp
  ...
  print_finish, parent_id=parent_id
endif
...
```

Written by: Eric E. Dors, 15 November 1995.

MODIFICATION HISTORY:

HISTORY:

Fri Jan 23 19:16:46 1998, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Removed top_widget input and created the parent_id ;
keyword parameter to contain the base id of the
calling widget if it exists. Also added the
noconfirm parameter to prevent the popping of the
confirmation dialog for deleting of the print file.

</PRE><P>
(See print_window.pro)<P>
<HR>

<H2>PRINT_WINDOW</H2>
[Previous Routine]
[Next Routine]
[List of Routines]
<PRE>
NAME:

PRINT_WINDOW

PURPOSE:

This function creates a dialogue box for printing information, the generated events can put IDL into a state to make files or send plots to a printer.

CATEGORY:

General Utility

CALLING SEQUENCE:

Result = PRINT_INFO_WINDOW

Result: 1 unless print operation was canceled in which case 0 is returned.

KEYWORD PARAMETERS:

xsize: Optional keyword to allow removing of device dependent parameters for operations such as tv.

ysize: Optional keyword to allow removing of device dependent parameters for operations such as tv.

color_t: Color table to use for color printing. (39 default)

color_g: The gamma factor to use for color printing. (1.0 default)

bw_t: Color table to use for black and white printing. (0 default)

bw_g: The gamma factor to use for black and white printing. (1.0 default)

parent_id: The widget handle to the controlling base widget. This is used so that informational dialog can be centered on that widget.

COMMON BLOCKS:

print_event: A common block to pass information between the event handler and the widget creation routine.

SIDE EFFECTS:

The current plot device is changed.

NOTE By calling print_finish, a device,/close and a set_plot, 'x' will be done to restore to the normal state.

EXAMPLE:

```
result=print_window(xsize=draw_x, ysize=draw_y)
```

```

...
if (result) then begin
  data plotting routines go here
  time_stamp
  ...
  print_finish, parent_id=parent_id
endif
...

```

Written by: Eric E. Dors, 15 November 1995.

MODIFICATION HISTORY:

Fri Jan 23 18:46:48 1998, Eric E. Dors
 <dors@gamma.physics.uiowa.edu>

Simplified the creation of the main base widget.

Changed print_window to a function instead of a procedure. Returns 0 if canceled, 1 otherwise.

Also added a parent_id parameter to contain the base id of the calling widget if it exists. This facilitates the centering of widget_message events.

```

</PRE><P>
<STRONG>(See print_window.pro)</STRONG><P>
<HR>

```

```

<A NAME="READ_PRINT_CONFIG">
<H2>READ_PRINT_CONFIG</H2></A>
<A HREF="#PRINT_WINDOW">[Previous Routine]</A>
<A HREF="#ROUTINELIST">[List of Routines]</A>
<PRE>

```

NAME:
 read_print_config

PURPOSE:

The purpose of this subroutine is to read in the printing configuration file called print_window.cfg. And return a list of variables set in the configuration file to the caller.

CATEGORY:

General Utility

CALLING SEQUENCE:

Result = read_print_config(varlist)

KEYWORD PARAMETERS:

top_widget: the id of the parent widget base if it exists

OUTPUTS:

varlist: A 2xn array holding variable [name, definition] of each of the n variables defined in the configuration file.

result: Return 1 on successful configuration file read, and 0 on failure.

PROCEDURE:

get_config_filename is used to find the configuration file, then it is opened and read line by line with get_cfg_line. Since the configuration file is perscribed to have comment lines beginning with ';' and variable setting lines which look like 'name=definition' we separate the names from the definitions and return the result as a 2xn dimensional array of [[name1,def1],[name2,def2],...,[namen,defn]].

EXAMPLE:

```
result = read_print_config(varlist)
```

Written by: Eric E. Dors, 9 September 1997.

MODIFICATION HISTORY:

Fri Jan 23 19:42:06 1998, Eric E. Dors
<dors@gamma.physics.uiowa.edu>

Added top_widget parameter for use with widget_message.

```
</PRE><P>  
<STRONG>(See print_window.pro)</STRONG><P>  
<HR>  
  
</body>  
</html>
```

File Attachments

- 1) [print_window.pro](#), downloaded 64 times
 - 2) [print_window.cfg](#), downloaded 64 times
 - 3) [print_window.html](#), downloaded 98 times
-