

24. GLOSSARY OF GLOBAL SYSTEM VARIABLES

As you can tell by now, there are many system variables in Medley that are useful to know. The following sections gather many of the important variables together into groups relating to directory searching, system flags, history lists, system menus, windows, and, of course, the catchall miscellaneous category.

Directories

DISPLAYFONTDIRECTORIES

Its value is a list of directories to search for the bitmap files for display fonts. Usually, it contains the `FONT` directory where you copies the bitmap files (see Chapter 16), and the current connected directory. The current connected directory is specified by the atom `NIL`. Here is an example value of `DISPLAYFONTDIRECTORIES`.

A screenshot of a text editor window. The title bar at the top reads "* SEdit DISPLAYFONTDIRECTORIES Package: INTERLISP". The text area contains three lines of code: ("{DSK}<usr<local<lde<Fonts>display>presentation>" followed by "{DSK}<usr<local<lde<Fonts>display>publishing>" and "{DSK}<usr<local<lde<Fonts>display>printwheel>").

```
* SEdit DISPLAYFONTDIRECTORIES Package: INTERLISP
("{DSK}<usr<local<lde<Fonts>display>presentation>"
 "{DSK}<usr<local<lde<Fonts>display>publishing>"
 "{DSK}<usr<local<lde<Fonts>display>printwheel>")
```

Figure 24.1. Value for the Atom `DISPLAYFONTDIRECTORIES`

INTERPRESSFONTDIRECTORIES

Is set to a list of directories to search for the font width files for InterPress fonts.

DIRECTORIES

This variable is bound to a list of the directories you will be using (see Figure 24-2). The system uses this variable when it is trying to find a file to load. It checks each directory in the list, until the file is found. `NIL` in list means to check the current connected directory.

LISPUSERSDIRECTORIES

Its value is a list of directories to search for library package files.

Flags

DWIMIFYCOMPFLG

This flag, if set to `T`, will cause all expressions to be completely dwimified before the expression is compiled (see Chapter 9). In this state, when the system does not recognize a function of keyword, it will compare the word to a system maintained list to determine whether the word is a macro, `CLISP` word, or misspelled user-defined variable.

An example of swinifying before compilation is to convert an `IF` call to a `COND`. before they are compiled. Undwimified expressions can cause inaccurate compilation. This flag is set by the system to `NIL`. Normally, you want this set to `T`. For more information on DWIM, refer to the *IRM*.

SYSPRETTYFLAG

When set to `T`, all lists returned to the executive window are pretty printed. This flag is originally set by the system to `NIL`.

CLISPIFTRANFLG

When set to `T`, keeps the `IF` expression, rather than the `COND` translation in your code.

PRETTYTABFLG

When set to `T`, the pretty printer puts out a tab character rather than several spaces to try to make code align. If `NIL`, it uses space characters instead.

FONTCHANGEFLG

If `NIL`, then when pretty printing no font changes will happen (e.g., a smaller font for comments, bold for clip words, and so forth). The default is the atom `ALL`, so different fonts are used where appropriate.

AUTOBACKTRACEFLG

There are many possible values for this variable. They affect when the back trace window appears with the break window, and how much detail is included in it. The values of this variable include:

- `NIL`, its initial value. The back trace window is not brought up when an error is generated, until you open it yourself.
- `T`, which means that the back trace `BT` window is opened for error breaks
- `BT!` brings up a back trace window with more detail, `BT!`, window for error breaks
- `ALWAS` brings up a backtrace `BT` window for both error breaks, and breaks caused by calling the function `BREAK`
- `ALWAYS!` brings up a backtrace window with more detail, `BT!`, for both error breaks and breaks caused by calling the function

NOSPELLFLG

Is initially bound to `NIL`, so that `DWIM` tries to correct all spelling errors, whether they are in a form you just typed in or within a function being run. If the variable is `T`, then no spelling correction is performed. This variable is automatically reset to `T` when you are compiling a file. If it has some other non-`NIL` value, then spelling correction is only performed on type-in.

History Lists

LISPMXHISTORY

Originally set to the list `(NIL 0 30 100)`, with the following argument interpretation. The `NIL` is the list (implemented as a circular queue) to which the top level commands append. `0` is the current prompt number. `30` is the maximum length of the history list. `100` is the highest number used as a prompt. This is a system maintained list used by the programmers assistant commands `REDO`, `UNDO`, `FIX`, and `??` use to retrieve past function calls.

To delete the history list, reset the variable `LISPMXHISTORY` to its original value of `(NIL 0 30 100)`.

Setting this variable to `NIL` disables all the programmers assistant features.

EDITHISTORY

This is also set to `(NIL 0 30 100)`, and has the same description as `LISPMXHISTORY`. This list allows you to `UNDO` edits. You reset this the same way as `LISPMXHISTORY`.

System Menus

System menus are all bound to global variables and are easy to modify. If the menu name is set to the `NIL` value, the menu will be recreated using an items list bound to a global variable.

To change a system menu, edit the items list bound to the appropriate global variable (system menus use this items list with the default `WHENSELECTEDFN`), then set the value of the name to `NIL`. The next time you need the menu, it will be created from the items list you just edited. The names of system menus and the items lists follow.

BackgroundMenu

This is the variable bound to the menu this displays when you press the right button in the grey background area of the screen.

BackgroundMenuCommands

This list is used for the list of `ITEMS` for the background menu when it is created.

WindowMenu

This is the variable bound to the default window menu displayed when the right mouse button is pressed inside of a window.

WindowMenuCommands

This is the list of `ITEMS` for the WindowMenu.

BreakMenu

The menu displayed when the middle mouse button is pressed in a break window.

BreakMenuCommands

The list of `ITEM` for the BreakMenu.

Windows

PROMPTWINDOW

Global name of the prompt window.

T

Although the value `T` has several meanings (such as universal `TRUE`), it also stands for the standard output stream. As this is usually the executive window, it may be used as the name for the TTY Window at the top level. Mouse processes have their own TTY Windows. A reference to the window `T` in a mouse driven function (e.g., a `WHENSELECTEFN`, Chapter 12) will open a TTY Window for Mouse.

Miscellaneous

CLEANUPOPTION

This is a list of options that you set to automate clean-up after a work session. Example options are listing files, or recompilation. You will want to keep this set to `NIL` until you become comfortable with the machine.

FILELST

The list of all the files you loaded.

SYSFILES

The list of all the files you loaded for the `SYSOUT` file.

INITIALS

An atom you can bind to your name. If bound, the editor will add your name, in addition to the date, in the editor comment at the beginning of each function.

FIRSTNAME

If this variable is set, the system will use it to greet you personally when you log on to your machine.

INITIALSLST

A list of elements of the form (USERNAME . INITIALS) or (USERNAME FIRSTNAME INITIALS). This list is used by the function GREET to set your INITIALS, and your FIRSTNAME when you log in.

#CAREFULCOLUMNS

An integer. PRETTYPRINT estimates the number of characters in an atom, instead of computing it, for efficiency. Unfortunately, for very long atom names, errors can occur. #CAREFULCOLUMNS is the number of columns from the right within which PRETTYPRINT should compute the number of characters in each atom, to prevent these errors. Initially this is set to zero. PRETTYPRINT never computes the number of characters in an atom. If you set it to 20 or 30, when PRETTYPRINT comes within 20 or 30 columns of the right of the window, it will begin computing exactly how many characters are in each atom. This will prevent errors.

DWIMWAIT

Bound to the number of seconds DWIM should wait before it uses the default response, FIXSPELLDEFAULT, to answer its question.

FIXSPELLDEFAULT

Bound to either Y or N. Its value is used as the default answer to questions asked by DWIM that you don't answer in DWIMWAIT seconds. It is initially bound to Y, but is rebound to N when DWIMIFYing.

\TimeZoneComp

This is the global variable set to the absolute value of the time offset from Greenwich. For EST, \TimeZoneComp should be set to 5.