NAME

shelltool – run a shell (or other program) in an OpenWindows terminal window

SYNOPSIS

shelltool [**–C**] [**–B** *boldstyle*] [**–I** *command*] [*generic-tool-arguments*]
[*program* [*arguments*]]

DESCRIPTION

shelltool is a standard *OpenWindows* facility for shells or other programs that use a standard tty-based interface.

When invoked, **shelltool** runs a program, (usually a shell) in an interactive terminal emulator based on a tty subwindow. Keyboard input is passed to that program. In the OpenWindows version of **shelltool,** a restricted pop-up menu is available from the main display area that allows you to enable scrolling. Selecting the **Enable Scrolling** option from the restricted menu gives **shelltool** the full functionality of the **cmdtool** window, including a larger pop-up menu from which to select options. Selecting **Disable Scrolling** from the pop-up submenu will return **shelltool** to its original state.

OPTIONS

–C Redirect system console output to this **shelltool**.

–B *boldstyle* Set the style for displaying bold text to *boldstyle*. *boldstyle* can be a string specifying one of the choices for the **term.boldstyle** default, see **Defaults Options**, below, or it may be a numerical value for one of those choices, from 0 to 8, corresponding to the placement of the choice in the list.

–I *command* Pass *command* to the shell. SPACE characters within the command must be escaped.

generic-tool-arguments
shelltool accepts the generic tool arguments listed in **xview**(7).

USAGE

.Xdefaults File Options

You can specify a number of defaults using the options in the **.Xdefaults** file that affect the behavior of **shelltool**. The ones of interest are those that begin with **text**,**term**, or **keyboard**. See **xview**(7) for more detailed information.

The Terminal Emulator

The tty subwindow is a terminal emulator. Whenever a tty subwindow is created, the startup file **~/.ttyswrc** is read for initialization parameters that are specific to the tty subwindow.

The .ttyswrc File

The command format for this file is:

#	Comment.
set *variable*	Turn on the specified *variable*.
mapi *key text*	When *key* is typed pretend *text* was input.
mapo *key text*	When *key* is typed pretend *text* was output.

The only currently defined *variable* is **pagemode**. *key* is one of L1-L15, F1-F15, T1-T15, R1-R15, LEFT, or RIGHT on SPARC keyboards. On x86 keyboards *key* is one of F1-F12, End, PgDn, PgUp, HOME, LEFT, or RIGHT (see note below). *text* may contain escapes such as \E, \n, ^X, etc. (ESC, RETURN, and CTRL-X, respectively). See **curs_terminfo**(3X) for the format of the string escapes that are recognized. Note: **mapi** and **mapo** may be replaced by another keymapping mechanism in the future.

When using the default xserver keyboard tables, the keys L1, LEFT, RIGHT, BREAK, R8, R10, R12, and R14 on a SPARC keyboard cannot be mapped in this way; they send special values to the tty subwindow. Also, when using the default xserver keyboard tables, L1-L10 are now used by XView. Likewise, on an x86 keyboard, the keys LEFT, RIGHT, UP, and DOWN cannot be mapped this way. See **kbd**(1) for more information on how to change the behavior of the keyboard.

It is possible to have terminal-based special escape sequences. These escape sequences may also be sent by typing a key appropriately mapped using the **mapo** function described above. The following functions pertain to the tool in which the tty subwindow resides, not the tty subwindow itself.

\E[1t	– open
\E[2t	– close (become iconic)
\E[3t	– move, with interactive feedback
\E[3;TOP;LEFTt	– move, to **TOP LEFT** (pixel coordinates)
\E[4t	– stretch, with interactive feedback
\E[4;HT;WIDTHt	– stretch, to **HT WIDTH** size (in pixels)
\E[5t	– front
\E[6t	– back
\E[7t	– refresh
\E[8;ROWS;COLSt	– stretch, to **ROWS COLS** size (in characters)
\E[11t	– report if open or iconic by sending **\E[1t** or **\E[2t**
\E[13t	– report position by sending **\E[3;TOP;LEFTt**
\E[14t	– report size in pixels by sending **\E[4;HT;WIDTHt**
\E[18t	– report size in characters by sending **\E[8;ROWS;COLSt**
\E[20t	– report icon label by sending **\E]Llabel\E**
\E[21t	– report tool header by sending **\E]llabel\E**
\E]ltext\E	– set tool header to text
\E]Ifile\E	– set icon to the icon contained in file**;** file must be in *iconedit* output format
\E]Llabel\E	– set icon label to label
\E[>OPT;...h	– turn SB OPT on (**OPT** = 1 => pagemode), for example, **\E[>1;3;4h**
\E[>OPT;...k	– report **OPT**; sends **\E[>OPTl** or **\E[>OPTh** for each **OPT**
\E[>OPT;...l	– turn **OPT** off (**OPT** = 1 => pagemode), for .B \E[>1;3;

See the *Solaris User's Guide* for an example of using this facility.

shelltool Windows

The window created by **shelltool** is based on the ttysw package. This package provides a simple character-based terminal emulator interface. The user is given a prompt at which to type commands and pop-up menus from which to select command options.

shelltool windows support cursor motions, using the **/usr/share/lib/terminfo** entry called **sun-cmd**. Command windows automatically set the **TERM** environment variable to **sun-cmd**. So, if you **rlogin**(1) to a machine that does not have an entry for **sun-cmd** in its **/usr/share/lib/terminfo** file, the error message '**Type sun-cmd unknown**' results. To rectify this, type the command '**set TERM=sun**'. Programs written using the **curses**(3X) library packages will work in a command window, but programs hard-coded for **sun**-type terminals may not. When supporting a program that performs cursor motions, the command window automatically takes on the characteristics of a tty window (as with **shelltool**). When that program terminates or sleeps, the full command window functionality is restored.

The shelltool Menu

The **shelltool** window menu is called the **Term Pane** menu and contains the following options and their submenus:

Enable Page Mode
Enables page mode within **shelltool .**

Copy Places the highlighted text on the clipboard.

PastePuts the contents of
pointed to by the cursor.

Scrolling Enables scrolling within **shelltool .**

EXAMPLES

The following aliases can be put into your **~/.cshrc** file:

```
# dynamically set the name stripe of the tool:
alias header 'echo  -n  "\ E]l"\ !*\ E\\'
# dynamically set the label on the icon:
alias iheader 'echo -n "\ E]L"\ !*\ E\\'
# dynamically set the image on the icon:
alias icon 'echo -n "\ E]I"\ !*\ E\\'
```

FILES

~/.ttyswrc
/usr/openwin/lib/.ttyswrc
/usr/openwin/bin/shelltool
/usr/demo/*

SEE ALSO

cmdtool(1), **more**(1), **xview**(7), **rlogin**(1), **kbd**(1), **curs_terminfo**(3X)

Solaris User's Guide