NAME

resize − set environment and terminal settings to current xterm window size

SYNOPSIS

resize [**−v** | **−u** | **−c**] [**−s** [*row col*]]

DESCRIPTION

Resize prints a shell command for setting the appropriate environment variables to indicate the current size of *xterm* window from which the command is run.

Resize determines the command through several steps:

- first, it finds the name of the user's shell program. It uses the **SHELL** variable if set, otherwise it uses the user's data from /etc/passwd.
- then it decides whether to use Bourne shell syntax or C-Shell syntax. It uses a built-in table of known shells, which can be overridden by the **−u** and **−c** options.
- then *resize* asks the operating system for the terminal settings. This is the same information which can be manipulated using *stty*.
- then *resize* asks the terminal for its size in characters. Depending on whether the "**−s** option is given, *resize* uses a different escape sequence to ask for this information.
- at this point, *resize* attempts to update the terminal settings to reflect the terminal window's size in pixels:
  - if the **−s** option is used, *resize* then asks the terminal for its size in pixels.
  - otherwise, *resize* asks the operating system for the information and updates that after ensuring that the window's dimensions are a multiple of the character height and width.
  - in either case, the updated terminal settings are done using a different system call than used for *stty*.
- then *resize* updates the terminal settings to reflect any altered values such as its size in rows or columns. This affects the values shown by *stty*.
- finally, *resize* generates shell commands for setting the environment variables, and writes that to the standard output.

OPTIONS

The following options may be used with *resize*:

−c This option indicates that C shell commands should be generated even if the user's current shell does not appear to use C shell syntax.

−s [*rows columns*]

This option indicates that Sun console escape sequences will be used instead of the VT100-style *xterm* escape codes. If *rows* and *columns* are given, *resize* will ask the *xterm* to resize itself using those values.

Both of the escape sequences used for this option (first to obtain the window size and second to modify it) are subject to *xterm*'s **allowWindowOps** resource setting. The window manager may also choose to disallow the change.

The VT100-style escape sequence used to determine the screen size always works for VT100-compatible terminals. VT100s have no corresponding way to modify the screensize.

−u This option indicates that Bourne shell commands should be generated even if the user's current shell does not appear to use Bourne shell syntax.

−v This causes *resize* to print a version number to the standard output, and then exit.

Note that the Sun console escape sequences are recognized by XFree86 *xterm* and by *dtterm*. The *resize* program may be installed as *sunsize*, which causes makes it assume the **−s** option.

The *rows* and *columns* arguments must appear last; though they are normally associated with the **−s** option, they are parsed separately.

ENVIRONMENT

SHELL — Unless overridden by the **−c** option, *resize* determines the user's current shell by

- first checking if **$SHELL** is set, and using that,
- otherwise *resize* looks in the password file (/etc/passwd).

Generally Bourne-shell variants (including *ksh*) do not modify **$SHELL**, so it is possible for *resize* to be confused if one runs *resize* from a Bourne shell spawned from a C shell.

After determining the user's shell, *resize* checks the shell's name against a table of known shell names. If it does not find the name in its table, *resize* will use C shell syntax for the generated commands to set environment variables.

TERM — *Resize*'s generated shell command sets this to "xterm-new" if not already set.

TERMCAP — *Resize*'s generated shell command sets this variable on systems using termcap, e.g., when *resize* is linked with the *termcap* library rather than a *terminfo* library. The latter does not provide the complete text for a termcap entry.

COLUMNS, LINES — *Resize*'s generated shell command sets these variables on systems using terminfo. Many applications (including the curses library) use those variables when set to override their screensize.

FILES

/etc/termcap — for the base termcap entry to modify.

~/.cshrc — user's alias for the command.

EXAMPLES

For *resize*'s output to take effect, *resize* must either be evaluated as part of the command line (usually done with a shell alias or function) or else redirected to a file which can then be read in. From the C shell (usually known as */bin/csh*), the following alias could be defined in the user's *.cshrc*:

```
% alias rs 'set noglob; eval `resize`'
```

After resizing the window, the user would type:

```
% rs
```

Users of versions of the Bourne shell (usually known as */bin/sh*) that don't have command functions will need to send the output to a temporary file and then read it back in with the "." command:

```
$ resize > /tmp/out
$ . /tmp/out
```

AUTHORS

Mark Vandevoorde (MIT-Athena), Edward Moy (Berkeley)
Thomas Dickey (invisible-island.net).

SEE ALSO

use_env(3x)
csh(1), stty(1), tset(1)
xterm(1)