

NAME

dired – file directory editor

SYNOPSIS

dired [**-g**[*gid-or-groupname*]] [**-s**[**r**][*cnsrw*]] [**-u**[*uid-or-username*]] [**-v**] [**-w**[**f**|**h**]**number**]]
 [*directory-list*|*file-list*]

DESCRIPTION

dired displays a long-form **ls**(1) directory listing on the screen of a display terminal and allows you to ‘edit’ and peruse that listing by moving up and down, deleting, undeleting, editing, and displaying entries. The **emacs**(1) text editor provides a similar capability in an editor window. The shell *TERM* variable should be set to the standard string which the Berkeley *termcap* library uses for distinguishing terminals. With no argument, the connected directory is used. With only one argument, if that argument is a directory, it is used. With multiple arguments, (or a single non-directory argument) the argument(s) are interpreted as filenames. **dired** then types ‘Reading’ and gets information about the various files/directories in your specification. This may take a short while (depending on how many you give it), so it types one period (.) after the word ‘Reading’ for every 10 files it has gathered information about. With this, you can keep track of its progress. Interrupts, hangups, and the like are disabled since your terminal is put into a special mode that is only changed when you quit with the *q* command.

The format of the screen is as follows: each line represents a file (or directory), the name of which is right-most. From left the fields are: mode, link count, owner, size, write date and name. See **ls**(1) for a description of what each of these mean. You move up and down the column immediately left of the filename. The bottom half of the screen is used for displaying files via the **type** command. If there are too many files to all fit on one window, more windows are allocated. The **f** and **b** commands can be used for stepping forward and backward windows. The last screen line is used as an ‘echo’ line for displaying error messages and reading arguments. It also displays the full directory name if ‘direding’ a directory. When in split screen mode, the divider serves also as a ‘linear indicator’ showing where the current window is relative to the entire list of files. The symbols ‘(’ and ‘)’ denote the window. Square brackets replace ‘(’ and/or ‘)’ when the window is the first and/or last window. A single ‘o’ is used to represent the window when the window size is small compared to the total number of files.

OPTIONS

Command-line options, which are inherited by recursive invocations of **dired**, are:

-g[*gid-or-groupname*]

Show only files belonging to the specified group id number or group name. If the number or name is omitted, then the group id of the current user is assumed.

-s[**r**][*c*|**n**|**s**|**r**|**w**]

Forward or reverse sort by *creation date*, *name*, *size*, *read date*, or *write date* respectively. “Normal sort” is the order most often desired; it is largest first for size, and most recent for read and write date sorts. The default is to sort by name.

-u[*uid-or-username*]

Show only files belonging to the specified user id number or user name. If the number or name is omitted, then the user id of the current user is assumed.

-v Show the program version on *stderr*, and quit immediately.

-w[**f**|**h**]**number**

Use **number** lines for the directory index window, reserving the other half for quick file display. **f** means use the full screen for the index. **h** means use half of the screen for the index. **h** is the default.

KEYBOARD COMMANDS

Commands consist of single characters, with any necessary arguments prompted for, and displayed in the echo line. Several commands take an optional non-negative integer count argument, as in **vi**(1).

Here are the keyboard commands that control **dired**, organized into several categories:

Help

?, h	Display a help message summarizing the keyboard commands.
Quitting dired	
a	Abort out of the current directory. No deletions are done.
A	Abort completely out of dired , with no deletions.
q	Exit dired , displaying files marked for deletion and requiring confirmation before deleting them. If no confirmation is given (typing anything other than y), dired goes back to its display.
Q	Quit dired , with no deletions.
Actions on files	
!	Prompt for a system command to invoke. The command is executed, and confirmation is required before returning to the display. All % characters in the command are replaced with the full pathname of the current entry, and all # chars are replaced with just the trailing filename component (what you see on the screen).
.	Repeat the previous ! shell command, substituting the current entry for any special chars (%#) in the original command.
e	Run the editor defined in the <i>EDITOR</i> environment variable upon the current file. If <i>EDITOR</i> is not defined, vi (1) is used. However, if the current file is a directory, it is not edited, but rather, dired forks a copy of itself upon that directory. In this manner, you can examine the contents of that directory and thus move down the directory hierarchy.
m	Run the program defined by the <i>PAGER</i> environment variable program on the current entry. If <i>PAGER</i> is not defined, run more (1).
P	Print the current file on the lineprinter.
t	Type the file to the terminal; this is considerably faster than starting an editor on the file. In two-window mode, the bottom window is used, pausing after each screenful. The type-out may be interrupted by <control C> or q .
T	Same as t , but without any pauses at end-of-screen.
Deletion and undeletion	
d	Mark the current entry for deletion. Upon exit and confirmation (or re-reading using the 'R' command), this entry will be deleted. <i>Warning:</i> this includes directories! If it is a directory, everything in it and underneath it will be removed.
D	Mark <i>all</i> files for deletion. ('U' undoes this).
~	Mark files with names ending in ~ for deletion.
#	Mark files with names beginning with # for deletion.
u	Undelete the current entry, if it was previously marked for deletion.
U	Cancel all deletion requests.
Display	
<space>	Re-print the directory path name.
c	Refresh the current line.
\	Toggle between split-screen mode and full-screen mode.
<control G>	Show the current file number, the total number of files, and the percentage through the file. Useful in full screen mode when there is no linear indicator.

l, <control L>

Refresh the current window.

p

Display the full path name of the current file; embedded control characters are shown with graphics.

R

Re-read the directory or file list. If files are marked for deletion, **direcd** will first ask for confirmation and then delete them before re-reading. This is useful after operations done during shell escapes (e.g. **chmod(1)**).

r, s

Sort the file list by various fields: creation date, name, read date, size, write date. Only the first letter (e.g. **c**, **n**, **r**, **s**, or **w**) is required after giving the **r** and **s** commands. **s** sorts in increasing alphabetic, decreasing size, newest to oldest dates. **r** reverses the sense of **s**. For the **n**, **s**, and **w** subcommands, the date field is the write date. For the **r** subcommand, the date field is the read date. Whenever a sort is done, you are positioned at the top of the list afterwards. A sort can be aborted via **<ESC>**.

Miscellaneous**<ESC>**

Cancel a sort or a count.

Moving around**<down arrow>, <lf>, ^N, j**

Step to the next file. If this crosses a window boundary, the next window is displayed with a one-line overlap. May be preceded by a count.

<up arrow>, ^-, k, <backspace>, ^P

Step to previous file. If this crosses a window boundary, the previous window is displayed with a one-line overlap. May be preceded by a count.

<PageDown>, <right arrow>, f

Go forward a window, leaving a one-line overlap. May be preceded by a count.

<PageUp>, <left arrow>, b

Go backward a window, leaving a one-line overlap. May be preceded by a count.

<, [, (, {

Go to the start of the file list. With an argument *n*, go *n/10* of the way from the start.

>,],), }

Go to the end of the file list. With an argument *n*, go *n/10* of the way from the end.

E

Go up to the next higher level directory. In the case of an argument list of files to **direcd**, go to the parent of the directory which contains the current file.

G

Go to the file number given by the preceding count. With no count, go to the last file as in **vi(1)**.

M

Remember the current entry on the mark stack. You can later return to it with the **J** (jump) command.

J

Pop the top entry from the mark stack, and jump to it.

L

Pop the top entry from the view stack, and jump to it. Each screen display adds an entry to the view stack, except for views created by this command, so you can use **L** to display views in reverse order.

Searching**/<regular-expression>**

Locate a file matching the given regular expression, as defined in **re_comp(3)** and **re_exec(3)**. The search is in the forward direction. The regular expression is remembered for subsequent use by the **n** and **N** commands. If the regular expression is empty, the last one remembered is used.

- n** Find the next instance in the forward direction of the regular expression previously-defined by a `/` command.
- N** Find the next instance in the backward direction of the regular expression previously-defined by a `/` command.

DIAGNOSTICS

The error messages should be self-explanatory.

The exit codes returned to the invoking program are as follows:

- 0** success (normal termination)
- 1** failure (usually accompanied by an explanatory message)
- 2** failed to open `termcap` file
- 3** unknown terminal type
- 69** user requested abort with `A` or `Q`, or else editor invocation failed

FILES

- `/usr/local/bin/dired` Executable program.
- `/usr/local/src/dired` Source directory.
- `/usr/local/bin/dired.hlp` Help file for `?` and `h`.

SEE ALSO

emacs(1), **ls(1)**, **more(1)**, **re_comp(3)**, **re_exec(3)**, **vi(1)**.

AUTHORS

Stuart Mclure Cracraft.

Enhancements by Jay Lepreau <lepreau@cs.utah.edu>.

Fixes and enhancements by Charles Hill.

More fixes and enhancements, and major revision of manual pages, by Nelson H. F. Beebe <beebe@math.utah.edu>, who is the current maintainer of **dired**.

Regular-expression support code by Tatu Ylonen <ylo@ngs.fi>.

AVAILABILITY

dired is freely available; its master distribution can be found at

`ftp://ftp.math.utah.edu/pub/misc`

in the file `dired-x.yy.tar.gz` where `x.yy` is the current version. Other distribution formats are usually available in the same location.

That site is mirrored to several other Internet archives, so you may also be able to find it elsewhere on the Internet; try searching for the string *dired* at one or more of the popular Web search sites, such as

`http://altavista.digital.com/`
`http://www.hotbot.com/`
`http://www.stpt.com/`
`http://www.yahoo.com/`

BUGS

Long lines sometimes screw up the **t** (type) display.

@@@ gam Sun Apr 1 8:00:11 PM PST 1984 original version