

CONTENTS

NAME	1
SYNOPSIS	1
DESCRIPTION	1
OPTIONS	2
INTERACTIVE COMMANDS	4
CONFIGURATION FILE ~/.vddiffrc	12
VDDIFF MODES	16
One-column diff mode	16
Two-column diff mode	16
One-column browse mode (bmode).....	17
Two-column browse mode (fmode).....	17
MOUSE SUPPORT	17
FILES	17
EXAMPLES	17
SEE ALSO	18

NAME

vddiff — text terminal directory diff tool

SYNOPSIS

```
vddiff [-u [RC file]] [-BbCcDEefgIiKlMmnoqRrVWXy] [-F file name pattern]
[-G file content pattern] [-t diff_tool] [-v view_tool] [directory_1|file_1]
[directory_2|file_2]
```

DESCRIPTION

vddiff stands for “Vim Directory DIFF”. It is a simple text terminal directory diff tool using vim(1) for the actual diff operation. (If vim is not installed, “diff *file_1 file_2* | less” is used instead. Other diff tools can be configured as well.) A file browser displays the names found in the directories given on the command line. Only plain files and symbolic links are compared. Unless option **-r** is used, directories are not compared before they are entered.

Compressed files are automatically decompressed into a temporary directory before the diff tool is started. These files are detected by their file name extension, the file contents are not (yet) checked. Currently the following file name extensions are supported: bz2, gz, tar, tbz, tgz, txz, xz, zip, and Z.

If one argument is a directory and the other is a non-directory, a diff is done between the non-directory and it's last path component within the directory. If **vddiff** is started with only one or no argument it behaves like a simple single directory view file browser (a mode called **bmode** whereas the normal mode is called **diff mode**). By use of marks files and directories can be compared in bmode. If less than two arguments are used in two-column mode kind of a file manager mode (called **fmode**) is started. This mode does not support global marks. All **vddiff** modes are explained in section **VDDIFF MODES**.

Symbols are printed in two columns before the names which have the following meaning:

First column (or middle bar in two-column mode): Difference type

- ‘ ’ Directory, equal files or different file types
- ‘!’ Different files or links (or directories with option **-r**)
- ‘<’ Files found in first directory only
- ‘>’ Files found in second directory only

- '=' Files have same i-node
- '-' Error
- 'X' Two-column mode: Different file type

Second column (or first column in two-column mode and **bmode**): File type

- '/' Directory
- ' ' Plain file
- '@' Symbolic link
- 'c' Character device
- 'b' Block device
- '=' Socket
- '|' FIFO
- '?' Unknown file type
- '!' Different file type

OPTIONS

- B** Force two-column file browser mode even if two arguments are given. Normally diff mode is started with two arguments. Even if it is possible to change to browse mode with key '#', it may take a long time until the (not needed) diff is done. So **-B** avoids diff mode at all.
- b** Disable color.
- C** Don't compare file contents, only test for file existence. If differences regarding the existing of files are of interest in a directory with many huge files, it may take very long to compare their contents. **-C** does any tests but always assumes that two existing files are equal.

Alternatively it is also possible to type '%' to ignore file contents for the further directory diff. ('%' should be pressed only short and one time. The comparison of the current file continues which may take a while, but further files are not compared.)
- c** Show only directories which exist in both trees and really different files. If also **-r** is used only those directories are shown which contain differences.
- d** Use "diff \$1 \$2 | less" as diff tool. (If a specific diff tool (other than vim -d) should be used in general, it is recommended to configure it with the RC file option **difftool**, see section **CONFIGURATION FILE ~/.vddiffrc** below.)
- E** Use extended regular expressions.
- e** Use basic regular expressions.
- F pattern** Display only file names which match *pattern*, which is normally a case ignoring basic regular expression. With **-E** extended regular expressions are enabled, with **-I** case-sensitive matching is used. The pattern match is only applied to non-directories. To display only directories which contain matching files add option **-r** (command "**vddiff -rF pattern directory**" works in bmode somehow similar to "**find directory -name pattern**"). Key 'E' toggles display of all files and directories or matching files only. The pattern can be removed with the ":nofind" command.
- f** Normally directories are displayed on top. With this option files are displayed first.
- G pattern** Display only files which contain *pattern* (which is normally a case ignoring basic regular expression). With **-E** extended regular expressions are enabled, with **-I** case-

sensitive matching is used. The pattern match is only applied to regular files. To display only directories which contain matching files add option **-r** (command “**vddiff -rG pattern directory**” works in bmode somehow similar to “**grep -r pattern directory**”). Key ‘E’ toggles display of all files and directories or matching files only. Option **-G** can be combined with option **-F**. For example

```
vddiff -rF '\.[1-9]$\ ' -G '/usr'
```

finds all manpage source files which contain absolut path names. Option **-G** can be used multiple times, in this case **all patterns** are required to match. This differs from `grep(1)`, where at least one pattern needs to match. But this can also be expressed as

```
-EG '<pattern1>|<pattern1>'.
```

For example

```
vddiff -rF '\.[1-9]$\ ' -IG '^.Dd' -EG '/(usr|etc|var)'
```

finds all manpage source files in **-mdoc** format which contain absolut path names. Options **-E**, **-e**, **-I**, and **-i** can be used again before each **-G** option to specify the kind of regular expression and the case sensitivity for the *pattern*. Further patterns can be added with the “:grep *pattern*” command, all patterns can be removed with the “:nogrep” command.

- g** Use **gvim** as diff and view tool.
- I** Use case-sensitive pattern match.
- i** Use case-insensitive pattern match.
- k** Use **tkdiff** as diff tool.
- l** Follow symbolic links.
- M** Normally flags **-W** and **-X** are only applied on single files operations. **-M** allows to use them on multiple file too.
- m** Normally directories are displayed on top. This is disabled with this option.
- N** When using the ‘s’ (run shell) commands one may forget that vddiff is already running and starts a new instance instead of typing ^D to return to vddiff. The tool checks this case and prevents a second invocation.
- n** This option suppresses the display of equal files.
- o** Hide files which are on one side only.
- q** Debug option: Print differing files and exit (similar to “diff -q”, but output is unsorted).
- R** Read-only mode: All file change operations (change name, permissions, owner, group; copy, move, delete) and function keys are disabled. To enable write mode, input “:e” or “:edit”. To only enable function keys, input “:set fkeys”.
- r** Recursively scan directories to detect differences in subdirectories. This allows to mark directories which contain differences. It increases the start time (due to disk I/O) since the full file tree is compared at begin. To only show different directories this option needs to be combined with **-c**. Pressing key ‘c’ enables to view all files in this mode.
- t diff_tool** Specify diff tool on the command line. The filenames to compare are appended to the given string. To include them into the string the symbolic names “\$1” and “\$2” can be

used (in any order), where “\$1” refers to the first and “\$2” to the second file. Note that the shell may require quoting as in

```
-t "diff \$1 \$2 | vim -R -"
```

-u [*filename*]

Skip reading the initialization file at start-up. If **-u** is used, it must be the very first option. Other used options need to begin with **-** again. If an optional *filename* is supplied, this file is read instead of the default initialization file `~/.vddiffrc`. *filename* needs to be separated with white space from **-u**.

-v Print version and exit.

-v *view_tool*

Specify view tool on the command line. The *view_tool* is appended to the given string. To include it into the string the symbolic name “\$1” can be embedded which is expanded to the filename.

-W Don’t ask for confirmation on file system operations.

-X Don’t ask for confirmation on execute.

-y Start in two-column mode. This is currently only supported if two arguments are given.

INTERACTIVE COMMANDS

‘Q’, **“qY”** Quit **vddiff**.

‘h’ or **‘?’** Display help. In help mode the following inputs are valid:

‘q’, **⟨LEFT⟩** Leave help mode.

‘+’, **‘j’**, **⟨DOWN⟩**
Scroll down.

‘-’, **‘k’**, **⟨UP⟩** Scroll up.

Mouse scroll wheel
Scroll list.

‘ ’, **⟨PAGE-DOWN⟩**
Scroll one screen down.

⟨BACKSPACE⟩, **⟨PAGE-UP⟩**
Scroll one screen up.

⟨CTRL-L⟩ Redraw screen.

⟨CTRL-L⟩ Refresh display. This may be necessary after another application had output text into the curses controlled display.

⟨TAB⟩ In **fmode**: Toggle column.

⟨UP⟩, **‘k’** or **‘-’**
Move cursor line up.

⟨DOWN⟩, **‘j’** or **‘+’**
Move cursor line down.

⟨LEFT⟩ Leave directory (one directory up).

⟨RIGHT⟩, **⟨ENTER⟩**, or double click
View file, enter directory or start diff tool. Compressed files and directories are unpacked

before the view or diff tool is started.

If a file is marked (with ‘m’), only **<RIGHT>** and double click can be used to normally view files or enter directories. **<ENTER>** starts a diff between the marked file and the selected file or directory.

<PAGE-UP> or <BACKSPACE>	Scroll one screen up.
<PAGE-DOWN> or <SPACE>	Scroll one screen down.
<HOME> or “1G”	Go to first file.
<END> or ‘G’	Go to last file.
<i>n</i> ‘G’	Go to line <i>n</i> .
‘ ’ <LEFT>	In two-column mode: Enlarge right column by 10 characters.
‘ ’ <RIGHT>	In two-column mode: Enlarge left column by 10 characters.
“ =”	In two-column mode: Make column widths equal.
<CTRL-w>	Toggle two-column mode.
‘/’	Search file in list by typing the begin of the filename. Searching is normally done case-insensitive. Set option noic to change this. Search mode is left when a file is selected (with <RIGHT> or <ENTER>).
“//”	Search with a basic regular expression for a filename. This can be configured with options noic (don’t ignore case), magic (use extended regular expressions), and nows (don’t wrap around when search hits top or bottom of the file list). Regex search mode is not left until ‘r’ is pressed. Previously entered search patterns are saved in a history, which can be accessed with the <UP> and <DOWN> keys.
“Sd”	Sort files with directories on top.
“Sm”	Sort files by name only (ignoring file type).
“SS”	Sort files by size with directories on top, smallest file first.
“St”	Sort files by modification time only, oldest first (ignoring file type).
‘H’	Put cursor to top line.
‘M’	Put cursor on middle line.
‘L’	Put cursor on bottom line.
“z <ENTER> ”	Put selected file to top.
“Z.”	Center selected file.
“z-”	Put selected file to bottom.
<CTRL-e>	Scroll one line down.
<CTRL-y>	Scroll one line up.

<CTRL-d>	Scroll half screen down.
<CTRL-u>	Scroll half screen up.
‘!’ or ‘n’	Toggle display of equal files.
‘c’	Toggle display of all files or only directories which exist in both trees and really different files. If option -r is used only those directories are shown which contain differences.
‘&’	Toggle display of files which are on one side only.
“&l”	Hide files which are on left side only.
“&r”	Hide files which are on right side only.
‘^’	Toggle display of files which are in both trees.
‘F’	Toggle following symbolic links.
‘E’	Toggle file name (-F) or file content (-G) filter.
“Ah”	Add scaled file size column. Does not show device major and minor numbers (future version will show this for “As”) Removes an existing byte size column. Can be enabled permanently with .vddiffrc option disp_hsize .
“Ag”	Add file group column. Can be enabled permanently with .vddiffrc option disp_group .
“Ap”	Add file permissions column. Can be enabled permanently with .vddiffrc option disp_perms .
“As”	Add file size column. Does not (yet) show device major and minor numbers. Removes an existing scaled size column.
“At”	Add file modification time column. Can be enabled permanently with .vddiffrc option disp_mtime .
“Au”	Add file owner column. Can be enabled permanently with .vddiffrc option disp_owner .
“Rh”	Remove scaled file size column.
“Rg”	Remove file group column.
“Rp”	Remove file permissions column.
“Rs”	Remove file size column.
“Rt”	Remove file modification time column.
“Ru”	Remove file owner column.
‘p’	Show current relative work directory.
‘a’	Show command line directory arguments.
‘f’	Show full path.
[n]“<<”	Copy from second to first tree. If <i>n</i> is given, <i>n</i> files starting from the current selection are copied.

If symbolic links in source and target are followed or not is toggled with ‘F’. Following links is signaled with letter F on the right side of the status line. Asks for a name for the

	copy if source and destination directory are the same.
[n]<>>	Copy from first to second tree.
“'<<”	Copy all files between the cursor and the local mark (inclusive) from second to first tree. After any ‘'’ command the cursor is set to the mark position if it had been below the mark.
“'>>”	Copy all files between the cursor and the local mark (inclusive) from first to second tree.
[n]‘C’	Copy to other side.
“'C”	Copy all files between the cursor and the local mark (inclusive) to other side.
[n]‘U’	Update files: Overwrite older files. Files with equal modification time and directories are ignored.
“'U”	Update all files between the cursor and the local mark (inclusive).
[n]“dd”	Delete file or directory. Does not follow symbolic links. vddiff does not warn if a directory to delete is not empty. If <i>n</i> is given, <i>n</i> files starting from the current selection are deleted. Key <DELETE> can be used instead of the key sequence “dd”.
[n]“dl”	Delete file or directory in first tree. (Does not follow symbolic links.)
[n]“dr”	Delete file or directory in second tree. (Does not follow symbolic links.)
“'dd”	Delete all files between the cursor and the local mark (inclusive). Does not follow symbolic links.
“'dl”	Delete all files between the cursor and the local mark (inclusive) in first tree. (Does not follow symbolic links.)
“'dr”	Delete all files between the cursor and the local mark (inclusive) in second tree. (Does not follow symbolic links.)
[n]‘T’	Move file or directory to the other file tree. Does a rename if source and destination directory are the same. If used in diff mode it is tested if there is any selected file which can be moved automatically (does exist on one side only). When there is no such file the tool waits until ‘l’ or ‘r’ is pressed.
[n]“Tl”	Diff mode: Move file or directory to left file tree.
[n]“Tr”	Diff mode: Move file or directory to right file tree.
“'T”	Move all files between the cursor and the local mark (inclusive) to the other file tree.
“'Tl”	Diff mode: Move all files between the cursor and the local mark (inclusive) to left file tree.
“'Tr”	Diff mode: Move all files between the cursor and the local mark (inclusive) to right file tree.
[n]‘@’	Create symbolic link in other column to selected file in active column. Asks for a name for the link if source and destination directory are the same. If used in diff mode it is tested if there is any selected file which does exist on one side only. When there is no such file the tool waits until ‘l’ or ‘r’ is pressed.
[n]“@l”	Diff mode: Create symbolic link in left file tree to file in right file tree.

[n]“@r”	Diff mode: Create symbolic link in right file tree to file in left file tree.
“'@”	Create symlinks in other column to all files between the cursor and the local mark (inclusive) in active column.
“'@l”	Diff mode: Create symlinks in left file tree to all files between the cursor and the local mark (inclusive) in right file tree.
“'@r”	Diff mode: Create symlinks in right file tree to all files between the cursor and the local mark (inclusive) in left file tree.
“en”	Rename file.
“eln”	Rename file in first tree.
“ern”	Rename file in second tree.
[n]“ep”	Change permissions of file. If following symbolic links is not enabled the command is ignored for symbolic links. If decimal digit <i>n</i> is given, the mode of <i>n</i> files starting from the current selection is set.
[n]“elp”	Change permissions of file in first tree.
[n]“erp”	Change permissions of file in second tree.
“'ep”	Change permissions of all files between the cursor and the local mark (inclusive).
“'elp”	Change permissions of all files between the cursor and the local mark (inclusive) in first tree.
“'erp”	Change permissions of all files between the cursor and the local mark (inclusive) in second tree.
[n]“eu”	Change owner of file. If following symbolic links is not enabled the command is ignored for symbolic links. Previously entered user names are saved in a history, which can be accessed with the <UP> and <DOWN> keys. If decimal digit <i>n</i> is given, the owner of <i>n</i> files starting from the current selection is set.
[n]“elu”	Change owner of file in first tree.
[n]“eru”	Change owner of file in second tree.
“'eu”	Change owner of all files between the cursor and the local mark (inclusive).
“'elu”	Change owner of all files between the cursor and the local mark (inclusive) in first tree.
“'eru”	Change owner of all files between the cursor and the local mark (inclusive), in second tree.
[n]“eg”	Change group of file. If following symbolic links is not enabled the command is ignored for symbolic links. Previously entered group names are saved in a history, which can be accessed with the <UP> and <DOWN> keys. If decimal digit <i>n</i> is given, the group of <i>n</i> files starting from the current selection is set.
[n]“elg”	Change group of file in first tree.
[n]“erg”	Change group of file in second tree.
“'eg”	Change group of all files between the cursor and the local mark (inclusive).
“'elg”	Change group of all files between the cursor and the local mark (inclusive) in first tree.

“'erg”	Change group of all files between the cursor and the local mark (inclusive), in second tree.
‘P’	Create directory (bmode and fmode only).
“P1”	Create directory in left tree.
“Pr”	Create directory in right tree.
‘.’	Repeat last file system or function key command. For commands using marks and the ‘.’ key the number of files is saved. This number is applied on a ‘.’ command, starting from the cursor position (the mark is not relevant then).
‘m’	Mark file or directory. This can be used to compare files or directories which had been renamed or compressed in one file tree. Marks are defined globally and can be used to compare files from different file hierarchy depths. They can only be undefined with the ‘r’ key (or redefined with the ‘m’ key). As long as the current directory is not left, the marked file is highlighted with either a bold font or a blue background. (In fmode marks are cleared if the directory is left or if any external command is executed.) If a mark is set, only <RIGHT> and double click can be used to normally view files or change to other directories. <ENTER> starts a diff between the marked and the selected file or directory.
‘V’, <INSERT> , mouse button 3	Set multiple marks. If no local mark is set (using ‘m’), ‘V’ toggles the mark of the current line and moves the cursor to the next line. If a local mark is set, the marks of all lines from the line with the local mark to the current line are inverted. Unless other marks multiple marks are removed with ‘u’ instead of ‘r’.
“VG”	Toggle mark of all files from the cursor position to the last line.
“1GVG”	Toggle mark of all files. (vim uses “gg” instead of “1G”, but this is not a traditional vi command.)
‘r’	Remove mark, edit line, or regex search. Does not remove multiple marks, use ‘u’ instead.
‘b’	Test for binary difference between selected and marked file. Compressed files are unpacked but compressed archive files are compared directly.
‘y’	Copy file path(s) to edit line. If a ‘\$’ command is entered later, this file path can be used to build a sh(1) command.
‘Y’	Copy file paths to edit line in reverse order.
‘\$’	Enter shell command. If paths had been copied to the edit line before using the ‘y’ or ‘Y’ command, the shell command can be prepended by pressing <HOME> and then entering the command. Predefined strings can be inserted by pressing a function key. The work directory is the directory where vdiff had been started. (In bmode and fmode the work directory is always the current view directory.) Each entered command is saved in a history. The keys <UP> and <DOWN> fetch other history entries. The shell to be used (default “sh”) can be configured with the sh option.
[n ’] <F1> – <F12>	Define string which can be inserted later with this function key when entering a sh command using ‘\$’. This string is usually the name of a UNIX tool. Regularly used strings can be set using the RC file fkey command.

If the string begins with a ‘\$’ followed by at least one space (‘ ’) it is treated as shell command itself, which is applied to a selected file(s). If that function key is pressed later, a dialog opens to ask if the command should be executed or the function key should be redefined. The filename is appended to the saved string. To embed it, “\$1” and “\$2” can be used, as in “\$ nroff \$1 | less”.

If *n* is given, the command is applied to *n* files starting from the current selection. If ‘!’ is prepended instead, the command is applied to all files between the cursor and the local mark (inclusive).

If the string starts with ‘!’ instead of ‘\$’, **<ENTER>** must be pressed after running the command. This allows to check the command’s output.

For strings starting with ‘\$’ or ‘!’ the user has to confirm the command execution. If the string starts with ‘#’ instead the command is started immediately.

To save a command for a function key which does not take the selected filename as argument, add a space followed by an octothorpe (“#”) at the end of the string.

If no diff tool which supports merging is available, files can be merged manually by defining two function keys for editing the left and right side file with “\$ vi \$1” and “\$ vi \$2”.

‘l’	List strings which had been defined for a function key.
‘u’	Update file list. Also used to remove multiple marks (‘r’ can’t be used to remove them).
‘s’	Open shell in the directory of the selected file. Does not work in diff mode when both files exist (use “sl” or “sr” instead). The shell to open can be set with the shell option, else it is read from the user’s entry in /etc/passwd.
“sl”	Open shell in left directory tree.
“sr”	Open shell in right directory tree.
‘o’	Open file (instead of diff tool).
“ol”	Open left file or directory. Plain files are opened with less(1) by default. If no diff tool which supports merging is available, files can be merged manually opening them with “ol” or “or” and then pressing ‘v’ in less which opens vi.
“or”	Open right file or directory.
‘v’	View raw file contents.
“vl”	View raw left file contents.
“vr”	View raw right file contents.
‘:’	Enter command: !shell_command Enter shell command.
cd	bmode and fmode only: Change to home directory.
cd path	bmode and fmode only: Change to directory <i>path</i> . <i>path</i> may be absolute or relative. ‘~’, “~user” and environment variables are expanded but not (yet) auto-completed. Other path components are auto-completed on pressing <TAB> . This does only work when the cursor is at the end of

	the input string.
<code>e[dit]</code>	Allow file change operations and function keys.
<code>find pattern</code>	Display only filenames which match <i>pattern</i> .
<code>nofind</code>	Remove filename pattern.
<code>grep pattern</code>	Display only files which contain <i>pattern</i> .
<code>nogrep</code>	Remove file content pattern.
<code>marks</code>	List jump marks.
<code>q, qa</code>	Quit vddiff.
<code>set all</code>	Display the current setting of the changable options.
<code>set fkeys</code>	Enable function keys.
<code>set nofkeys</code>	Disable function keys.
<code>set ic</code>	Set case-insensitive match.
<code>set noic</code>	Set case-sensitive match.
<code>set magic</code>	Use extended regular expressions.
<code>set nomagic</code>	Use basic regular expressions.
<code>set recursive</code>	Use recursive diff, find and grep operations.
<code>set norecursive</code>	Use directory local diff, find and grep.
<code>set ws</code>	File name searches wrap around top and bottom.
<code>set nows</code>	File name searches don't wrap around top and bottom.
<code>vie[w]</code>	Read-only mode: Disable file change operations and function keys.
Previously entered commands are saved in a history, which can be accessed with the <UP> and <DOWN> keys. If possible, commands and their arguments are auto-completed on pressing <TAB> . To e.g. input “set recursive” it is sufficient to type	
	<code>:s<TAB>r<TAB></code>
<code>#</code>	Toggle between diff mode and two-column browse mode (fmode).
<code>=</code>	In fmode : Set path of active column to path of the other column.
<code>%</code>	Toggle ignoring file contents for further diff operations. This is useful if only differences regarding the file existence are of interest. Any time this function is disabled, the currently displayed directories are compared again. <code>%</code> can be pressed during this directory compare (short and only one time) to abort it. Note that an active file diff is not interrupted. This may take some time in case of a huge file.
<code>Da</code>	Add current directory to a persistent list. Items on this list can be selected later using the

	“D1” command.
“D1”	Show persistent directory list. In this list the following inputs are valid:
	‘q’, <LEFT> Close directory list.
	‘+’, ‘j’, <DOWN> Move cursor down.
	‘-’, ‘k’, <UP> Move cursor up.
	‘ ’, <PAGE-DOWN> Scroll one page down.
	<BACKSPACE>, <PAGE-UP> Scroll one page up.
	“1G”, <HOME> Go to first line.
	‘G’, <END> Go to last line.
	<i>n</i> ‘G’ Change into directory <i>n</i> .
	<ENTER>, <RIGHT> Change into directory.
	“dd”, <DELETE> Delete entry.
	Mouse button 1 Select entry.
	Mouse button 1 double click Change into directory.
	Mouse button scroll wheel Scroll list.
	‘H’ Place cursor in top line.
	‘M’ Place cursor in middle line.
	‘L’ Place cursor in bottom line.
	‘/’ Start regex search.
	‘n’ Find next matching pattern.
	‘N’ Find previous matching pattern.
	‘r’ Finish regex search.
	<CTRL-L> Rebuild screen.
<CTRL-g>	Print cursor line number and number of files.
“,”	Jump to previous cursor position.
<i>n</i> ‘m’	Set jump mark. <i>n</i> is a number between 0 and 31. (In contrast to vi a number instead of a letter is used and ‘m’ is behind instead of before the number.) Jump marks save the file position, not the file name. If the position of the file in the list changes, the mark points to an other file.

<code>n''</code>	Jump to mark <i>n</i> .
<code>''1''</code>	List jump marks.
<code>'W'</code>	Toggle to always wait for <ENTER> after running an external tool. This can be useful for debugging purposes.

CONFIGURATION FILE `~/.vddiffrc`

Permanent non-default options can be set in the file `~/.vddiffrc`, which has a very simple (*not* .exrc like) syntax. The elements in this file may be separated with spaces, tabs or line breaks. Line breaks are not required, everything can be written into one long line. Also spaces or tabs are not required (outside quoted strings), when every element is on it's own line. Everything following a '#' (outside quoted strings) to the end of the line is a comment.

Quoted strings can be split over several lines by placing a backslash ('\\') at the end of a line. No character is allowed to follow this backslash on the same line, not even a white space character. All occurrences of a backslash followed by a new line character (i.e. a line break) are removed before storing the quoted string.

To embed a double quote sign (") into a quoted string, precede it with a backslash. This backslash is removed before storing the string. All other backslashes are preserved.

`twocolumn` Start in two-column mode. This is currently only supported if two arguments are given.

`difftool string`

Configure *string* as diff tool. If *string* contains spaces it needs to be quoted with '"' at begin and end. The two filenames are appended to this string. If the filenames need to be before a pipe symbol ('|'), the symbols "\$1" and "\$2" can be embedded into *string* (in any order). These symbols are expanded to the respective filename.

Other possible diff tools are `colordiff(1)`, `mgdiff(1)`, or `tkxcd(1)`.

`difftool bg string`

Start *string* as a background process (don't block **vdiff** while executing *string*). **bg** is ignored if one of the files to be compared is compressed. Since they are decompressed into a temporary directory which is removed after starting the diff tool, the file may be removed before the tool reads it.

`viewtool string`

Configure *string* as view tool. If *string* contains spaces it needs to be quoted with '"' at begin and end. The filename is appended to this string. If the filename need to be before a pipe symbol ('|') the symbol "\$1" can be embedded into *string*, which is expanded to the filename.

`viewtool bg string`

Start *string* as a background process. **bg** is ignored if the file to be viewed is compressed. Since it is decompressed into a temporary directory which is removed after starting the view tool, the file may be removed before the tool reads it. For example

```
viewtool bg "xterm -e less"
```

displayes every file to view in a separate window while not blocking the file browser.

`ext extension string`

Configure view tool *string* for filenames ending with ".*extension*". *extension* is compared case-insensitive. If *extension* is an integer number it needs to be enclosed in double quotes (") as in

```
ext "1" "tbl $1 | neqn | nroff -mandoc | less"
```

because the parser expects a string (and not a number) after the keyword `ext`.

`ext extension bg string`

Start *string* as a background process.

`alias name string`

If a pipe of tools (like the `nroff` pipe above) or a command with options (like e.g. “`mplayer -idx --`” should be applied to many file types, an alias *name* can be defined for such a *string*. The `ext` command can then use this alias *name* instead the full command *string*, for example

```
alias mandoc "tbl $1 | neqn | nroff -mandoc | less"
alias mplayer "mplayer -idx --"
```

```
ext "1" mandoc
ext "2" mandoc
ext "3" mandoc
```

```
ext mts mplayer
ext ts mplayer
```

It is also possible to use an alias for defining another alias:

```
alias mplayer "mplayer -idx --"
alias audio mplayer
```

```
ext mpg mplayer
ext mp3 audio
```

`skipext extension`

When checking a filename extension skip *extension* at the end of the filename. E.g.

```
skipext old
```

would remove “.old” from the end of each filename before checking the extension.

`fkey number string`

Define *string* which can be inserted later with this function key when entering a `sh` command using ‘\$’. This string is usually the name of a UNIX tool. *number* must be between 1 and 12 (inclusive).

If *string* begins with a ‘\$’ followed by at least one space (the string needs to be enclosed in double quotes in this case), it defines a shell command to be executed for the selected file when pressing the function key.

If *string* starts with ‘!’ instead of ‘\$’, **<ENTER>** must be pressed after running the command. This allows to check the command’s output.

To save a command for a function key which does not take the selected filename as argument, add a space followed by an octothorpe (“ #”) at the end of the string.

`mono` Disable colors.

`followlinks`

Follow symbolic links.

noequal Display differences only, hide equal files.

real_diff Show only directories which exist in both trees and really different files.

recursive Recursively scan file tree to detect and mark directories which contain differences. This increases the start time. To show only directories with differences additionally set option **real_diff**. To view all files when in this mode key ‘c’ can be used.

noic Searching for a filename with ‘/’ or “//” is normally done case-insensitive. Case-sensitive search can be configured with this option.

magic For searching with “//” normally basic regular expressions are used. Use of extended regular expressions is configured with this option.

nows Searching for a filename with “//” normally wraps around top and bottom of the file list. This behaviour is disabled with this option.

include *string*
 Include RC file *string*. If *string* does not start with ‘/’, \$HOME is prepended. ‘~’ and environment variables are *not* expanded. (The include command does only work if vddiff had been build with a lex version which supports buffer switching.)

filesfirst Display directories at the end instead on top.

mixed Display files and directories mixed.

dir_color *integer*
 Set color for directories. Default is 3 (yellow).

diff_color *integer*
 Set color for different files. Default is 1 (red).

link_color *integer*
 Set color for symbolic links. Default is 5 (magenta).

left_color *integer*
 Set color for files found in first directory only. Default is 6 (cyan).

right_color *integer*
 Set color for files found in second directory only. Default is 2 (green).

unknown_color *integer*
 Set color for unknown file types. Default is 4 (blue).

normal_color *integer*
 Set color for normal text. Default is 7 (white).

bg_color *integer*
 Set background color. Default is 0 (black). E.g.
 bg_color 4 mark_color 7 0
 changes the background color to blue (and the normally blue mark color to black).

cursor_color *foreground background*
 Set the cursor color. *foreground* and *background* must be given as integers. Default is 0 7 (black on white).

mark_color *foreground background*
 Set color for marks. *foreground* and *background* must be given as integers. De-

fault is 7 4 (white on blue).

mmrk_color *foreground background*
Set color for multiple marks. *foreground* and *background* must be given as integers. Default is 0 3 (black on yellow).

error_color *foreground background*
Set color for diff errors. *foreground* and *background* must be given as integers. Default is 7 1 (white on red).

histsize *integer*
Set history size to *integer*. Default is 100. A size less than 2 disables the history (at least the current command line and one previous entry need to be saved to use this feature).

scale Show file size in human-readable format.

disp_perms Add file permissions column. Can be removed with “Rp”

disp_owner Add file owner column. Can be removed with “Ru”

disp_group Add file group column. Can be removed with “Rg”

disp_hsize Add scaled file size column. Can be removed with “Rh”

disp_mtime Add file modification time column. Can be removed with “Rt”

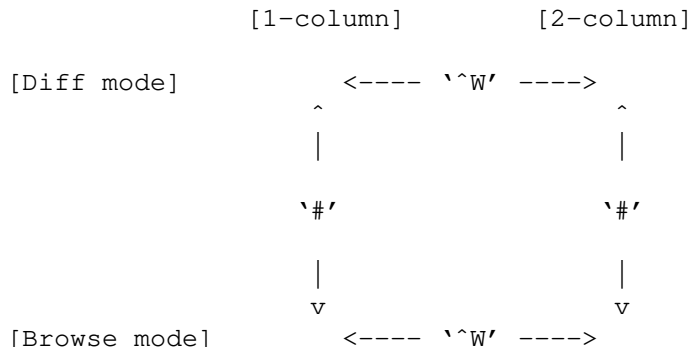
shell *string*
Set shell for the ‘s’, “sl” and “sr” command. Default is the user’s login shell. *string* may be a full path or just the shell name. In the latter case \$PATH is searched to determine the full path.

sh *string* Set shell for the ‘\$’ and function key commands, .gz and .bz2 uncompressing and any use of the shell special characters |, &, ;, <, >, (,), `, \, ", ', [, #, and ~. Default is “sh”. *string* may be a full path or just the shell name. In the latter case \$PATH is searched to determine the full path.

readonly Read-only mode: All file change operations (change name, permissions, owner, group; copy, move, delete) and function keys are disabled. Function keys get enabled if a **fkey** keyword is read after **readonly**. To avoid this, **readonly** can be placed at the end of the RC file.

VDDIFF MODES

vdiff has four modes: A diff mode and a browser mode, both have an one-column and a two-column submode. The following table shows the transition between these modes: Key **<CTRL-w>** toggles between one-column and two-column mode and key ‘#’ toggles between diff mode and browse mode:



While the diff modes access files with their path name, the browse modes always set the current working directory to the active file list column. This is relevant if shell commands are applied which create files. Marks are cleared on any mode change except switching between one-column and two-column diff mode.

One-column diff mode

Diff mode is entered when two arguments are used. Key **<CTRL-w>** switches to two-column diff mode, key **#** switches to two-column browse mode.

Two-column diff mode

Two-column diff mode is equal to diff mode except for a different data presentation. Key **<CTRL-w>** switches to single-column diff mode, key **#** switches to two-column browse mode.

One-column browse mode (bmode)

Bmode is a simply file browser mode. It allows file and directory comparison by use of marks. Key **<CTRL-w>** switches to two-column browse mode.

Two-column browse mode (fmode)

Fmode is a simple two-column file manager mode. It allows comparison of files and directories by use of local marks. This mode does not have global marks. If a directory is left, an existing mark in this directory is deleted. Generally in any mode at most one mark is supported (except the multi-marks set with **V**). If a mark is defined, an existing mark (in either column) is deleted.

The symbolic variables **“\$1”** and **“\$2”** exchange when the cursor is in the right column. That means **“\$1”** is the file from the cursor column and **“\$2”** is the marked file in the other column.

Key **<CTRL-w>** switches to single-column browse mode, key **#** or a compare of a marked directory (or archive) with a directory (or archive) switches to two-column diff mode.

MOUSE SUPPORT

For mouse support currently the ncurses interface is used which means that ncurses is required if one wants to use the mouse. Furthermore at least ncurses version 6 is currently required for using the mouse scroll wheel.

If the middle bar can be moved with the mouse to resize the columns in two-column mode is terminal dependend. See section **PORTABILITY** in mouse(3ncurses). For example, to make this function best work in xterm, set **“TERM=xterm-1002”** (else usually a further mouse click is required to finalize the resize).

FILES

```
~/vddiffrc
    Read on start-up to set non-default options.

~/vddiffinfo
    Storage for persistant information.
```

EXAMPLES

To display only differing files and subdirectories which contain differing files in two-column mode enter

```
vddiff -c directory_1 directory_2
```

where **-c** hides equal files, files which exist in one directory only and directories which don't contain differing files. Pressing **c** toggles showing all files. **-r** enables recursive mode. This option increases the start time, if huge directories are compared. If this takes too long, **%** can be pressed to disable further file content comparison. **-y** enables two-column mode.

To only hide equal files enter

```
vddiff -nry directory_1 directory_2
```

Pressing 'n' toggles showing all files.

If a recursive diff is not required,

```
vddiff -y directory_1 directory_2
```

(optionally with option **-c** or **-n**) is much faster, since a diff is only done, when a directory is entered. The only disadvantage is, that also directories may be shown, which contain no differences (since the tool had not visited them yet).

The arguments of vddiff need not be directories. If `~/vddiffrc` is setup accordingly,

```
vddiff file.1.gz
```

opens a manual page,

```
vddiff file.mpg
```

plays a movie, etc. Compressed files are unpacked automatically.

```
vddiff file_1 file_2
```

starts **vimdiff**. Again, any argument may be a compressed file.

```
vddiff file.tgz
```

opens an archive,

```
vddiff file_1.tgz file_2.zip
```

compares two archives and

```
vddiff file.txz directory
```

compares an archive with a directory.

SEE ALSO

<http://n-t-roff.github.io/vddiff/vddiff.1.pdf>

This manual page as a PDF document with hyperlinks.

<http://n-t-roff.github.io/vddiff/vddiff.1.html>

This manual page as a HTML document with hyperlinks.