" vim: filetype=vifm :

" Sample configuration file for vifm (last updated: 16 March, 2017)

" You can edit this file by hand.

" The " character at the beginning of a line comments out the line.

" Blank lines are ignored.

" The basic format for each item is shown with an example.

" ------------------------------------------------------------------------------

" This is the actual command used to start vi. The default is vim.

" If you would like to use another vi clone such as Elvis or Vile

" you will need to change this setting.

set vicmd=nvim

" set vicmd=elvis\ -G\ termcap

" set vicmd=vile

" Trash Directory

" The default is to move files that are deleted with dd or :d to

" the trash directory. If you change this you will not be able to move

" files by deleting them and then using p to put the file in the new location.

" I recommend not changing this until you are familiar with vifm.

" This probably shouldn't be an option.

set trash

" This is how many directories to store in the directory history.

set history=100

" Automatically resolve symbolic links on l or Enter.

set nofollowlinks

" With this option turned on you can run partially entered commands with

" unambiguous beginning using :! (e.g. :!Te instead of :!Terminal or :!Te<tab>).

" set fastrun

" Natural sort of (version) numbers within text.

set sortnumbers

" Maximum number of changes that can be undone.

set undolevels=100

" If you installed the vim.txt help file set vimhelp.

" If would rather use a plain text help file set novimhelp.

set novimhelp

" If you would like to run an executable file when you

" press return on the file name set this.

set norunexec

" Use KiB, MiB, ... suffixes instead of K, M, ... for size.

set noiec

" Selected color scheme

colorscheme desert

" Format for displaying time in file list. For example:

" TIME\_STAMP\_FORMAT=%m/%d-%H:%M

" See man date or man strftime for details.

set timefmt=%m/%d\ %H:%M

" Show list of matches on tab completion in command-line mode

set wildmenu

" Display completions in a form of popup with descriptions of the matches

set wildstyle=popup

" Display suggestions in normal, visual and view modes for keys, marks and

" registers (at most 5 files). In other view, when available.

set suggestoptions=normal,visual,view,otherpane,keys,marks,registers

" Ignore case in search patterns unless it contains at least one uppercase

" letter

set ignorecase

set smartcase

" Don't highlight search results automatically

set nohlsearch

" Use increment searching (search while typing)

set incsearch

" Try to leave some space from cursor to upper/lower border in lists

set scrolloff=4

" Don't do too many requests to slow file systems

if !has('win')

set slowfs=curlftpfs

endif

" Set custom status line look

set statusline=" Hint: %z%= %A %10u:%-7g %15s %20d "

" ------------------------------------------------------------------------------

" :mark mark /full/directory/path [filename]

" :bmark! path tag1 [tag2 [tag3...]]

bmark! ~/Work/Projects/endpoints/pd/billing/endpoints/ billing-endpoints

bmark! ~/Work/Projects/web/applications/billing-webapp/ web

bmark! ~/Work/Projects/web/pd/billing/angular/app/services/ billing-services

bmark! ~/Stuff/paycorp\_lk/hnb-cartridge/ hnb-cartridge

" ------------------------------------------------------------------------------

" :com[mand][!] command\_name action

" The following macros can be used in a command

" %a is replaced with the user arguments.

" %c the current file under the cursor.

" %C the current file under the cursor in the other directory.

" %f the current selected file, or files.

" %F the current selected file, or files in the other directory.

" %b same as %f %F.

" %d the current directory name.

" %D the other window directory name.

" %m run the command in a menu window

command! df df -h %m 2> /dev/null

command! diff vim -d %f %F

command! zip zip -r %f.zip %f

command! run !! ./%f

command! make !!make %a

command! mkcd :mkdir %a | cd %a

command! vgrep vim "+grep %a"

command! reload :write | restart

" ------------------------------------------------------------------------------

" The file type is for the default programs to be used with

" a file extension.

" :filetype pattern1,pattern2 defaultprogram,program2

" :fileviewer pattern1,pattern2 consoleviewer

" The other programs for the file type can be accessed with the :file command

" The command macros %f, %F, %d, %F may be used in the commands.

" The %a macro is ignored. To use a % you must put %%.

" For automated FUSE mounts, you must register an extension with :file[x]type

" in one of following formats:

"

" :filetype extensions FUSE\_MOUNT|some\_mount\_command using %SOURCE\_FILE and %DESTINATION\_DIR variables

" %SOURCE\_FILE and %DESTINATION\_DIR are filled in by vifm at runtime.

" A sample line might look like this:

" :filetype \*.zip,\*.jar,\*.war,\*.ear FUSE\_MOUNT|fuse-zip %SOURCE\_FILE %DESTINATION\_DIR

"

" :filetype extensions FUSE\_MOUNT2|some\_mount\_command using %PARAM and %DESTINATION\_DIR variables

" %PARAM and %DESTINATION\_DIR are filled in by vifm at runtime.

" A sample line might look like this:

" :filetype \*.ssh FUSE\_MOUNT2|sshfs %PARAM %DESTINATION\_DIR

" %PARAM value is filled from the first line of file (whole line).

" Example first line for SshMount filetype: root@127.0.0.1:/

"

" You can also add %CLEAR if you want to clear screen before running FUSE

" program.

" Pdf

filextype \*.pdf zathura %c %i &, apvlv %c, xpdf %c

fileviewer \*.pdf pdftotext -nopgbrk %c -

" PostScript

filextype \*.ps,\*.eps,\*.ps.gz

\ {View in zathura}

\ zathura %f,

\ {View in gv}

\ gv %c %i &,

" Djvu

filextype \*.djvu

\ {View in zathura}

\ zathura %f,

\ {View in apvlv}

\ apvlv %f,

" Audio

filetype \*.wav,\*.mp3,\*.flac,\*.m4a,\*.wma,\*.ape,\*.ac3,\*.og[agx],\*.spx,\*.opus

\ {Play using ffplay}

\ ffplay -nodisp %c,

\ {Play using MPlayer}

\ mplayer %f,

fileviewer \*.mp3 mp3info

fileviewer \*.flac soxi

" Video

filextype \*.avi,\*.mp4,\*.wmv,\*.dat,\*.3gp,\*.ogv,\*.mkv,\*.mpg,\*.mpeg,\*.vob,

\\*.fl[icv],\*.m2v,\*.mov,\*.webm,\*.ts,\*.mts,\*.m4v,\*.r[am],\*.qt,\*.divx,

\\*.as[fx]

\ {View using ffplay}

\ ffplay -fs %f,

\ {View using Dragon}

\ dragon %f:p,

\ {View using mplayer}

\ mplayer %f,

fileviewer \*.avi,\*.mp4,\*.wmv,\*.dat,\*.3gp,\*.ogv,\*.mkv,\*.mpg,\*.mpeg,\*.vob,

\\*.fl[icv],\*.m2v,\*.mov,\*.webm,\*.ts,\*.mts,\*.m4v,\*.r[am],\*.qt,\*.divx,

\\*.as[fx]

\ ffprobe -pretty %c 2>&1

" Web

"filextype \*.html,\*.htm

" \ {Open with dwb}

" \ dwb %f %i &,

" \ {Open with firefox}

" \ firefox %f &,

" \ {Open with uzbl}

" \ uzbl-browser %f %i &,

"filetype \*.html,\*.htm links, lynx

" Object

filetype \*.o nm %f | less

" Man page

filetype \*.[1-8] man ./%c

fileviewer \*.[1-8] man ./%c | col -b

" Images

filextype \*.bmp,\*.jpg,\*.jpeg,\*.png,\*.gif,\*.xpm

\ {View in sxiv}

\ sxiv %f,

\ {View in gpicview}

\ gpicview %c,

\ {View in shotwell}

\ shotwell,

fileviewer \*.bmp,\*.jpg,\*.jpeg,\*.png,\*.gif,\*.xpm

\ convert -identify %f -verbose /dev/null

" OpenRaster

filextype \*.ora

\ {Edit in MyPaint}

\ mypaint %f,

" Mindmap

filextype \*.vym

\ {Open with VYM}

\ vym %f &,

" MD5

filetype \*.md5

\ {Check MD5 hash sum}

\ md5sum -c %f %S,

" SHA1

filetype \*.sha1

\ {Check SHA1 hash sum}

\ sha1sum -c %f %S,

" SHA256

filetype \*.sha256

\ {Check SHA256 hash sum}

\ sha256sum -c %f %S,

" SHA512

filetype \*.sha512

\ {Check SHA512 hash sum}

\ sha512sum -c %f %S,

" GPG signature

filetype \*.asc

\ {Check signature}

\ !!gpg --verify %c,

" Torrent

filetype \*.torrent ktorrent %f &

fileviewer \*.torrent dumptorrent -v %c

" FuseZipMount

filetype \*.zip,\*.jar,\*.war,\*.ear,\*.oxt,\*.apkg

\ {Mount with fuse-zip}

\ FUSE\_MOUNT|fuse-zip %SOURCE\_FILE %DESTINATION\_DIR,

\ {View contents}

\ zip -sf %c | less,

\ {Extract here}

\ tar -xf %c,

fileviewer \*.zip,\*.jar,\*.war,\*.ear,\*.oxt zip -sf %c

" ArchiveMount

filetype \*.tar,\*.tar.bz2,\*.tbz2,\*.tgz,\*.tar.gz,\*.tar.xz,\*.txz

\ {Mount with archivemount}

\ FUSE\_MOUNT|archivemount %SOURCE\_FILE %DESTINATION\_DIR,

fileviewer \*.tgz,\*.tar.gz tar -tzf %c

fileviewer \*.tar.bz2,\*.tbz2 tar -tjf %c

fileviewer \*.tar.txz,\*.txz xz --list %c

fileviewer \*.tar tar -tf %c

" Rar2FsMount and rar archives

filetype \*.rar

\ {Mount with rar2fs}

\ FUSE\_MOUNT|rar2fs %SOURCE\_FILE %DESTINATION\_DIR,

fileviewer \*.rar unrar v %c

" IsoMount

filetype \*.iso

\ {Mount with fuseiso}

\ FUSE\_MOUNT|fuseiso %SOURCE\_FILE %DESTINATION\_DIR,

" SshMount

filetype \*.ssh

\ {Mount with sshfs}

\ FUSE\_MOUNT2|sshfs %PARAM %DESTINATION\_DIR,

" FtpMount

filetype \*.ftp

\ {Mount with curlftpfs}

\ FUSE\_MOUNT2|curlftpfs -o ftp\_port=-,,disable\_eprt %PARAM %DESTINATION\_DIR,

" Fuse7z and 7z archives

filetype \*.7z

\ {Mount with fuse-7z}

\ FUSE\_MOUNT|fuse-7z %SOURCE\_FILE %DESTINATION\_DIR,

fileviewer \*.7z 7z l %c

" Office files

filextype \*.odt,\*.doc,\*.docx,\*.xls,\*.xlsx,\*.odp,\*.pptx libreoffice %f &

fileviewer \*.doc catdoc %c

fileviewer \*.docx, docx2txt.pl %f -

" TuDu files

filetype \*.tudu tudu -f %c

" Qt projects

filextype \*.pro qtcreator %f &

" Directories

filextype \*/

\ {View in thunar}

\ Thunar %f &,

" Syntax highlighting in preview

"

" Explicitly set highlight type for some extensions

"

" 256-color terminal

" fileviewer \*.[ch],\*.[ch]pp highlight -O xterm256 -s dante --syntax c %c

" fileviewer Makefile,Makefile.\* highlight -O xterm256 -s dante --syntax make %c

"

" 16-color terminal

" fileviewer \*.c,\*.h highlight -O ansi -s dante %c

"

" Or leave it for automatic detection

"

" fileviewer \*[^/] pygmentize -O style=monokai -f console256 -g

" Displaying pictures in terminal

"

" fileviewer \*.jpg,\*.png shellpic %c

" Open all other files with default system programs (you can also remove all

" :file[x]type commands above to ensure they don't interfere with system-wide

" settings). By default all unknown files are opened with 'vi[x]cmd'

" uncommenting one of lines below will result in ignoring 'vi[x]cmd' option

" for unknown file types.

" For \*nix:

" filetype \* xdg-open

" For OS X:

" filetype \* open

" For Windows:

" filetype \* start, explorer

" ------------------------------------------------------------------------------

" What should be saved automatically between vifm runs

" Like in previous versions of vifm

" set vifminfo=options,filetypes,commands,bookmarks,dhistory,state,cs

" Like in vi

set vifminfo=dhistory,savedirs,chistory,state,tui,shistory,

\phistory,fhistory,dirstack,registers,bookmarks,bmarks

" ------------------------------------------------------------------------------

" Examples of configuring both panels

" Customize view columns a bit (enable ellipsis for truncated file names)

"

" windo set viewcolumns=-{name}..,6{}.

" Filter-out build and temporary files

"

" windo filter! /^.\*\.(lo|o|d|class|py[co])$|.\*~$/

" ------------------------------------------------------------------------------

" Sample mappings

" Start shell in current directory

nnoremap s :shell<cr>

" Display sorting dialog

nnoremap S :sort<cr>

" Toggle visibility of preview window

nnoremap w :view<cr>

vnoremap w :view<cr>gv

" Open file in existing instance of gvim

nnoremap o :!gvim --remote-tab-silent %f<cr>

" Open file in new instance of gvim

nnoremap O :!gvim %f<cr>

" Open file in the background using its default program

nnoremap gb :file &<cr>l

" Yank current directory path into the clipboard

nnoremap yd :!echo %d | xclip %i<cr>

" Yank current file path into the clipboard

nnoremap yf :!echo %c:p | xclip %i<cr>

" Mappings for faster renaming

nnoremap I cw<c-a>

nnoremap cc cw<c-u>

nnoremap A cw

" Open console in current directory

nnoremap ,t :!xterm &<cr>

" Open editor to edit vifmrc and apply settings after returning to vifm

nnoremap ,c :write | edit $MYVIFMRC | restart<cr>

" Open gvim to edit vifmrc

nnoremap ,C :!gvim --remote-tab-silent $MYVIFMRC &<cr>

" Toggle wrap setting on ,w key

nnoremap ,w :set wrap!<cr>

" Example of standard two-panel file managers mappings

nnoremap <f3> :!less %f<cr>

nnoremap <f4> :edit<cr>

nnoremap <f5> :copy<cr>

nnoremap <f6> :move<cr>

nnoremap <f7> :mkdir<space>

nnoremap <f8> :delete<cr>

" faster bookmarks list

nnoremap b :bmarks<cr>

" easy create new file

nnoremap aa :touch newfile<cr>

" yank current directory path into the clipboard

nnoremap yd :!echo -n %d | xclip %i<cr>

" yank current file path into the clipboard

nnoremap yf :!echo -n %c:p | xclip %i<cr>

" yank current filename without path into the clipboard

nnoremap yn :!echo -n %c | xclip %i<cr>

" yank root of current file's name into the clipboard

nnoremap yr :!echo -n %c:r | xclip %i<cr>

" copy file to the current pane

nnoremap ch :!echo -n '-R' %c:p %c.new | xargs cp<cr>

" un-tar a file

nnoremap ut :!echo -n '-xzvf' %c:p | xargs tar<cr>

" Custom Commands

"----------------

COMMAND=ln=!ln -s %d/%f %D

" ------------------------------------------------------------------------------

" Various customization examples

" Use ag (the silver searcher) instead of grep

"

set grepprg=ag\ --line-numbers\ %i\ %a\ %s

" Add additional place to look for executables

"

" let $PATH=$HOME.'/bin/fuse:'.$PATH

" Block particular shortcut

"

"