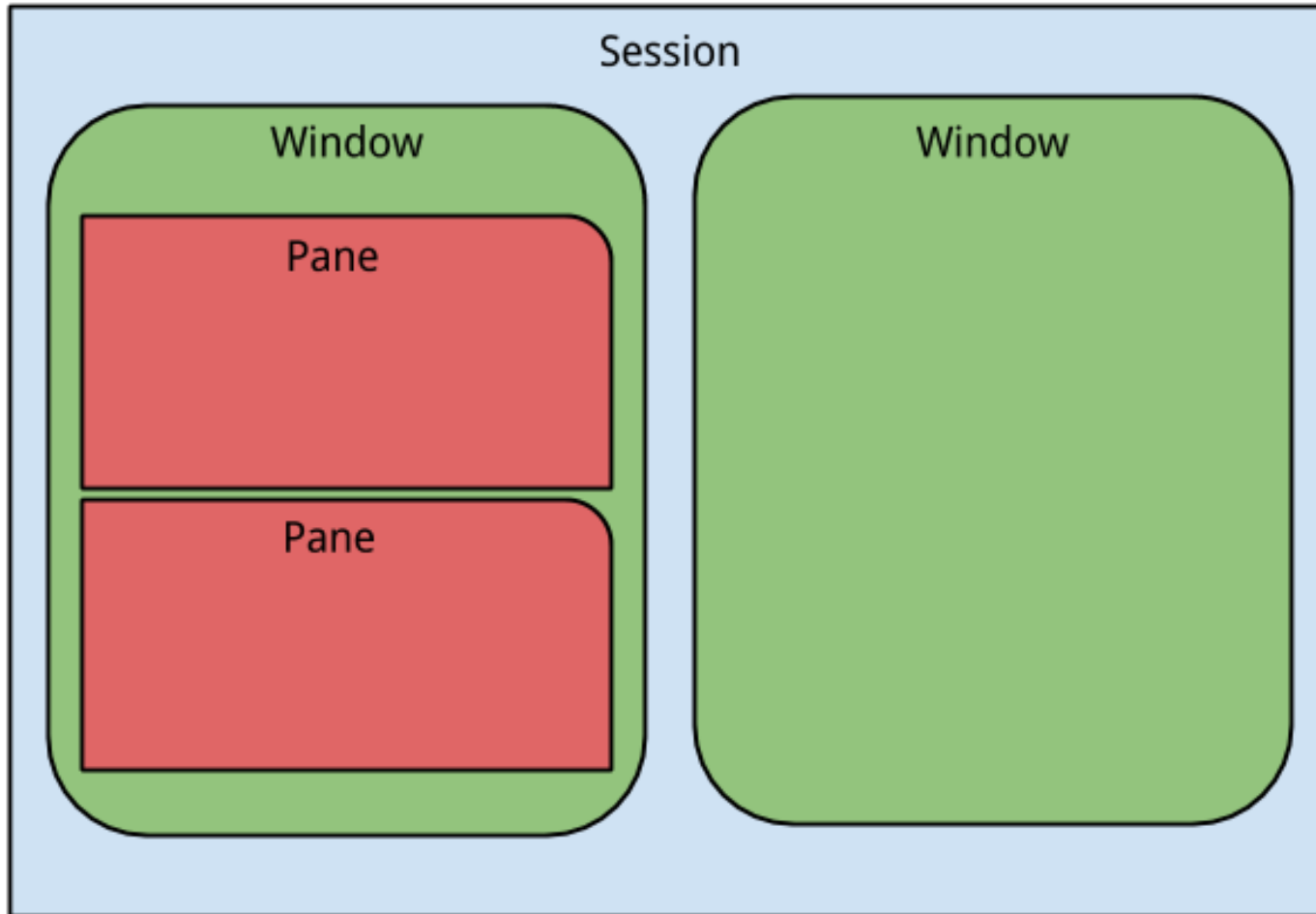


Consolidated zsh / tmux / vim Environment

TMUX

- Layout and status bar
- Hotkeys
 - Panes
 - Windows
 - Misc
- Copy mode
- Commands
- TMUX / VIM integration

TMUX overview



TMUX layout and status bar

```
Update /home/croysdal/.zshrc:
- [ -f ~/.fzf.zsh ] && source ~/.fzf.zsh
+ Added

Finished. Restart your shell or reload config file.
source ~/.bashrc # bash
source ~/.zshrc  # zsh

Use uninstall script to remove fzf.

For more information, see: https://github.com/junegunn/fzf

[2245] /home/croysdal/.fzf
croysdal/.fzf

drwxr-xr-x 1 croysdal croysdal 4.0K May 11 13:14 old
drwxr-xr-x 1 croysdal croysdal 4.0K Oct 20 14:46 perl
-rw-r--r-- 1 croysdal croysdal 100 Nov 2 16:23 README
drwxr-xr-x 1 croysdal croysdal 4.0K Oct 20 14:47 rebecca
drwxr-xr-x 1 croysdal croysdal 4.0K Oct 23 16:45 tcl
drwxr-xr-x 1 croysdal croysdal 4.0K Nov 2 16:24 tmp

[2314] /home/
bash_sc font ca tmp
bin Desktop docs GIT_README old README tcl

[2315] /home/croysdal
/home/croysdal █

1 Main
```

Pane 1

Pane 2

Window #

Active Window

1 interview 2* 2 python-

Session Title

Window Title

TMUX hot keys (overview)

- Hot Keys are normally 2 key strokes
 - Prefix (Ctrl+space) followed by hot key
 - Ctrl+space
 - Release Ctrl+space
 - Press Hot Key
- Sometimes, a single key combo is used
 - Alt-hjkl Move around panes
 - Shift-Arrows Move between windows
- Commonly used Hot Keys are marked in bold

TMUX overview

- Session

- Name of the tmux session running
- Good practice to use fvw desk as start of name
 - TODO: Automate this with a start_tmux command

- Windows

- These are the tabs in the tmux window
- Pane title can have these suffixes
 - *: Active, Z: Zoomed pane, Active output, Silent..

- Panes

- These are the split windows within a tmux window
- Can zoom a pane to the full Window with (Ctrl-space) Z

TMUX hot keys (panes)

Panes are similar to split windows in vim

Ctrl+space s	Split pane horizontally.
Ctrl+space v	Split pane vertically.
Ctrl+space z	Maximize pane
Ctrl+space hjkl	Switch pane (left/up/down/right)
Alt-hjkl	Switch pane (left/up/down/right)
Alt-arrows	Switch pane (left/up/down/right)
Ctrl+space o	Switch to the next pane
Ctrl+space q	Show pane numbers
Ctrl+space {	Move the current pane right
Ctrl+space }	Move the current pane left
Ctrl+space Ctrl+arrow	Hold Ctrl+arrow down to resize pane

TMUX hot keys (windows)

Windows are similar to tabs in vim

Ctrl+space c	Create a new window.
Shift-arrow	Move to left/right window
Ctrl+space n	Move to n ext (right) window
Ctrl+space p	Move to p revious (left) window
Ctrl+space [0-9]	Move to window #
Ctrl+space ,	Rename window (<i>comma</i>)
Ctrl+space	Move to previous window
Ctrl+space &	Kill the current window
Ctrl+space f	Jump to window by name

TMUX hot keys (misc)

Ctrl+space ?	List all key bindings
Ctrl+space \$	Rename the current session.
Ctrl+space d	Detach current client
Ctrl+space Ctrl-a	Monitor window for activity
Ctrl+space Ctrl-s	Monitor window for silence

TMUX copy mode

Copy mode is used for scrolling and searching.

- **Entering copy mode**

Ctrl+space b

Ctrl+space Ctrl+space

- **Hot keys used in edit mode**

Ctrl-b

Page Up

Ctrl-f

Page Down

?

Search Up

/

Search Down

Enter

Exit edit mode

v OR <space>

Start text select

Vi keys

Most vi keys work in copy mode

TMUX copy mode

```
Generate /home/croysdal/.fzf.bash ... OK
Generate /home/croysdal/.fzf.zsh ... OK

Do you want to update your shell configuration files? ([y]/n) y

Update /home/croysdal/.bashrc:
- [ -f ~/.fzf.bash ] && source ~/.fzf.bash
+ Added

Update /home/croysdal/.zshrc:
- [ -f ~/.fzf.zsh ] && source ~/.fzf.zsh
+ Added

Finished. Restart your shell or reload config file.
source ~/.bashrc # bash
source ~/.zshrc # zsh

Use uninstall script to remove fzf.

For more information, see: https://github.com/junegunn/fzf

[2245] /home/croysdal/.fzf
croysdal/.fzf
1 Main
```

In copy mode

Press Enter to exit copy mode

[0/28]

1 interview 1*Z 2 python-

TMUX commands

Start tmux

```
alias tmux='tmux -2 -u'
```

List existing tmux sessions

```
alias tml='tmux list-sessions'
```

Attach to existing tmux session

```
alias tma='tmux -2 -u attach -d -t'
```

Can re-attach even if xterm dies

```
tmux switch-client -t
```

```
tmux select-window -t
```

TMUX commands

tmux switch client w/ zsh command completion

```
[2277] /home/croysdal
/home/croysdal tmux switch-client -t
completing window
1 -- zsh* (1 panes) [80x23] [layout 575f,80x23,0,0,10] @6 (active)
completing sessions
0 First -- 1 windows (created Sat Nov 2 15:58:00 2019) [108x32] (attached)
1 Main -- 2 windows (created Sat Nov 2 16:23:06 2019) [79x23]
2 -- 1 windows (created Sun Nov 3 07:56:18 2019) [80x23] (attached)
```

TMUX and VIM integration

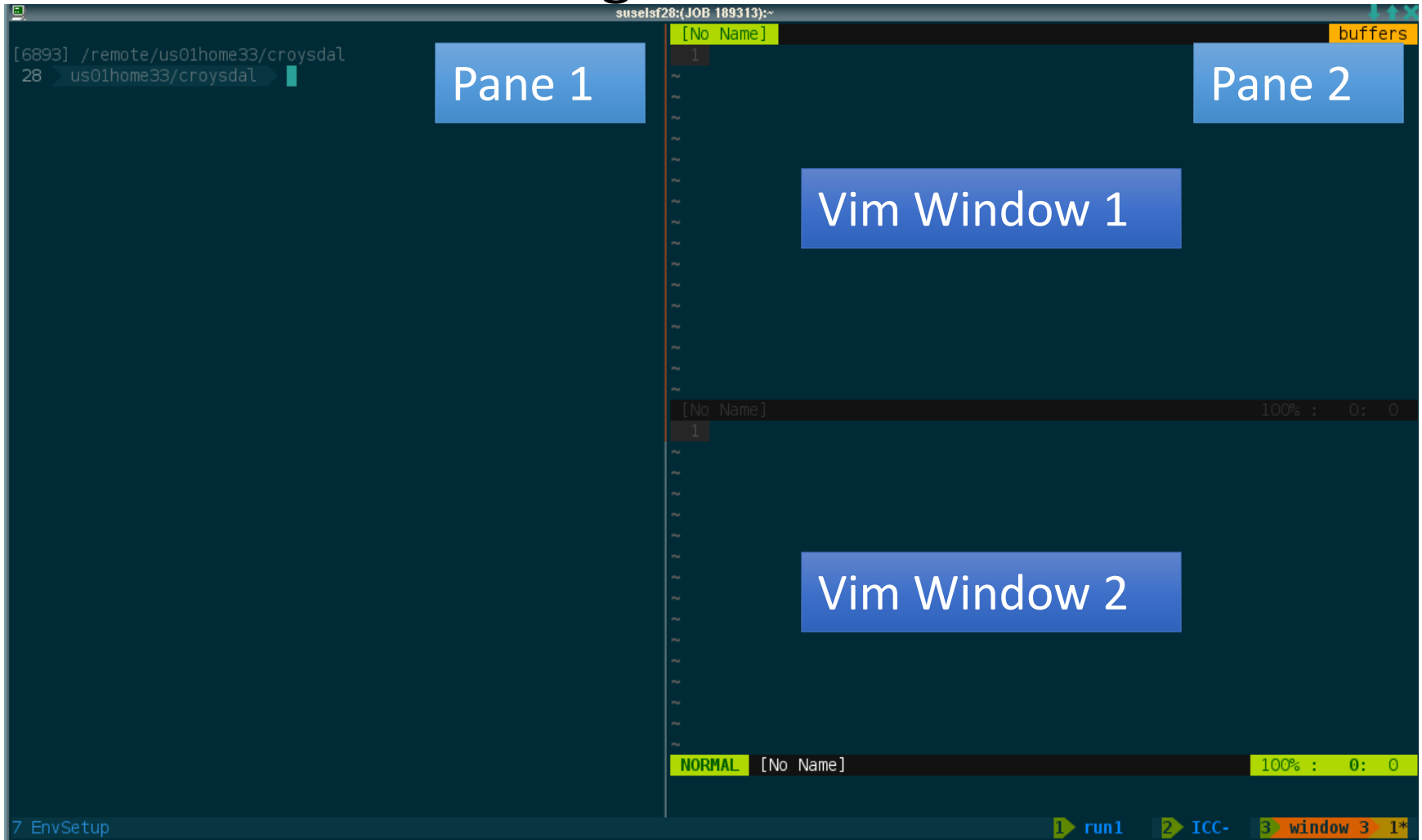
Same hot keys for panes/windows

Alt-J	Select pane or vim window up
Alt-K	Select pane or vim window down
Alt-H	Select pane or vim window left
Alt-L	Select pane or vim window right

Leveraged from:

<http://robots.thoughtbot.com/seamlessly-navigate-vim-and-tmux-splits>

TMUX and VIM integration



TMUX configuration

Location:

`~/.tmux.conf`

Sections of file

Key bindings

Scrollback / pastebuffer

Mouse mode

Pane splitting

Pane navigation

Colors / statusbar

dircolors

Location:

`~/.dir_colors.256`

Dir_colors optimized for 256 color terminal

Uses solarized dark colors

TMUX FAQ

- How do I scroll?

Ctrl-Space Ctrl-B to enter edit-mode

Ctrl-Space Ctrl-Space also works

Once in copy mode, use Ctrl-B/Ctrl-F or PgUp/PgDn

Search backwards with "?"

TMUX URLs

[Man page](#)

[TMUX Primer](#)

[TMUX copy mode](#)

<https://wiki.archlinux.org/index.php/tmux>

<http://blog.hawkhost.com/2010/06/28/tmux-the-terminal-multiplexer/>

Zsh

- Completion
 - Command, processes, host,
- Command prompt
- Functions & Aliases
- Persistent shell history
- oh-my-zsh

Command option completion

- `chmod` -

```
28 5.0.6/functions chmod -  
completing option  
--changes -c          -- report changes made  
--help              -- display help information  
--recursive -R        -- change files and directories recursively  
--reference          -- copy permissions of specified file  
--silent            --quiet -f -- suppress most error messages  
--verbose -v         -- output a diagnostic for every file processed  
--version           -- display version information  
completing privilege  
X -- execute only if executable to another  
g -- group's current permissions  
o -- other's current permissions  
r -- read  
s -- set uid/gid  
t -- sticky  
u -- owner's current permissions  
w -- write  
x -- execute
```

Context aware completion

- `chmod -R a`

```
28 5.0.6/functions chmod -R a
completing who
a -- all
g -- group
o -- others
u -- user
completing operator
\= - +
```

- `chmod -R a+`

```
28 5.0.6/functions chmod -R a+
completing privilege
X -- execute only if executable to another
g -- group's current permissions
o -- other's current permissions
r -- read
s -- set uid/gid
t -- sticky
u -- owner's current permissions
w -- write
x -- execute
```

- `Chmod -R a+r`

```
28 5.0.6/functions chmod -R a+r ../
completing file
../ functions/ help/ scripts/
```

Custom command completion

- Easy to add for gnu compliant commands (uses --help)

```
compdef _gnu_generic get_xtermcontrol
```

```
/home/croysdal xtermcontrol --get-  
get-bg          get-cursor    get-font        get-highlight   get-mouse-fg  
get-colorN      get-fg          get-geometry    get-mouse-bg    get-title  
completing option  
--get-bg        -- report background color  
--get-colorN    -- report N'th (0-15) color  
--get-cursor    -- report cursor color  
--get-fg        -- report foreground color  
--get-font      -- report font  
--get-geometry  -- report size and position  
--get-highlight -- report highlight color  
--get-mouse-bg  -- report mouse pointer background color  
--get-mouse-fg  -- report mouse pointer foreground color  
--get-title     -- report window title
```

Process completion

```
28 us01home33/croysdal kill 1768
```

```
completing process ID
```

```
1768 croysdal zsh
```

```
10185 croysdal zsh
```

```
10655 croysdal zsh
```

```
10924 croysdal perl.orig
```

```
11605 croysdal zsh
```

```
11619 croysdal zsh
```

```
12707 croysdal qrsh_starter
```

```
12718 croysdal csh
```

```
12851 croysdal xterm
```

```
12861 croysdal tcsh
```

```
13115 croysdal zsh
```

```
14641 croysdal zsh
```

```
15007 croysdal qrsh_starter
```

```
15014 croysdal csh
```


Path completion/expansion/replacement

- **Fuzzy completion**

```
cd /h/c/do  
<hit TAB>  
cd /home/croysdal/docs/
```

- **Error correction:**

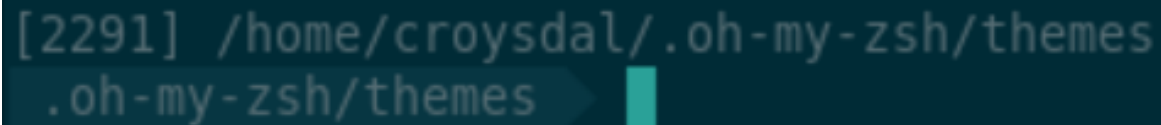
```
cd /home/croysdale/docs/  
<hit TAB>  
cd /home/croysdal/docs/
```

- **Path replacement**

```
cwd: /home/croysdal/docs/  
> cd docs perl  
cwd: /home/croysdal/perl
```

Dynamic command prompt

- [History #] Full_path
- machine > short_path

A terminal window with a dark background. The prompt is [2291] /home/croysdal/.oh-my-zsh/themes. The text is light blue. A teal cursor is at the end of the prompt.

```
[2291] /home/croysdal/.oh-my-zsh/themes
```

- Defined here:
~croysdal/.oh-my-zsh/themes/kevin.zsh-theme

z function

<https://github.com/rupa/z>

- c restrict matches to subdirectories of the current directory
- h show a brief help message
- l list only
- r match by rank only
- t match by recent access only
- x remove the current directory from the datafile

EXAMPLES

- z foo cd to most frecent dir matching foo
- z foo bar cd to most frecent dir matching foo, then bar
- z -r foo cd to highest ranked dir matching foo
- z -t foo cd to most recently accessed dir matching foo
- z -l foo list all dirs matching foo (by frecency)

z examples

```
[2320] /home/croysdal  
/home/croysdal ➤ z Doc
```

```
[2321] /mnt/c/Users/croys/Documents  
croys/Documents ➤ z .zsh
```

```
[2322] /home/croysdal/.zsh  
croysdal/.zsh ➤ z Pers
```

```
[2323] /mnt/c/Users/croys/Documents/Personal  
Documents/Personal ➤
```

Directory stack

- List with

<Control-t>

> dirs -v

> dstack

- Jump with

– cd ~#

```
Documents/Personal ➤ dirs -v
0      /mnt/c/Users/croys/Documents/Personal
1      ~/.zsh
2      /mnt/c/Users/croys/Documents
3      ~
4      ~/docs
5      /mnt/c/Users/croys/Documents/Personal/SNPS/Customers/CDS
6      ~/.oh-my-zsh/themes
```

Zsh-syntax-highlighting

- Dynamic syntax highlight command line
 - Quickly catch typos

```
[6800] /remote/us01home33/croysdal  
28 > us01home33/croysdal > ech
```

```
[6800] /remote/us01home33/croysdal  
28 > us01home33/croysdal > echo "this is a
```

```
[6800] /remote/us01home33/croysdal  
28 > us01home33/croysdal > echo "this is a test"
```

Expanded history

Expanded history with date stamps and command number

```
> alias h='fc -l -E -30'
```

#	ID	Date	Time	Command
				> h
2320		3.11.2019	09:12	z Doc
2321		3.11.2019	09:12	z .zsh
2322		3.11.2019	09:12	z Pers
2323		3.11.2019	09:13	dirs -v

Shared history

History is separate within each shell window

History is written to `.zsh_history` after every command

New windows start with history of all previous windows

Load history from all shell with `load_hist`

```
alias load_hist='fc -RI'
```

Useful if you want all history copied into existing window

Persistent shell history

- Keep full history of all commands in daily log files

Store in date stamped file:

`~/.zsh_logs/zsh-history-2015-02-04.log`

`date stamp, machine, tmux-session:window, dir, history`

`Wed Feb 4 20:10:43 PST 2015 suselsf28 : ICF (1) ICC : /remote/us01home33/croysdal 6730 ls`

- Uses `precmd` in: `~/.zsh/zshfunction`

```
function precmd() {  
  if [ "$(id -u)" -ne 0 ]; then  
    TMUX_WIN_PANE=$(tmux display-message -p '#S-#W')  
    FULL_CMD_LOG="$HOME/.zsh_logs/zsh-history-$(date "+%Y-%m-%d").log"  
    echo "$(date) `hostname` : ${TMUX_WIN_PANE} `pwd` ``\`history -1`" >>! ${FULL_CMD_LOG}  
  fi  
}
```

- Can `grep` files to find any command ever run...

History functions

- Search history:
 - `grep_hist_full` Search all history
 - `grep_hist_current_dir` Containing current working dir
 - `grep_hist_tmux_session` Matching TMUX session title
 - `grep_hist_tmux_window` Matching TMUX window title

```
[715] /remote/us01home33/croysdal
28 us01home33/croysdal grep_hist_tmux_session echo
Thu Feb 5 15:09:19 PST 2015 suselsf28 : hist_test-histwin2 /remote/us01home33/croysdal 709 echo "histwin2 cmd1"
Thu Feb 5 15:09:21 PST 2015 suselsf28 : hist_test-histwin2 /remote/us01home33/croysdal 710 echo "histwin2 cmd2"
Thu Feb 5 15:09:51 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 709 echo "histwin1 cmd1"
Thu Feb 5 15:09:56 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 710 echo "histwin1 cmd2"
Thu Feb 5 15:11:06 PST 2015 suselsf28 : hist_test-histwin2 /remote/us01home33/croysdal 713 grep_hist_tmux_window histwin2 echo
Thu Feb 5 15:11:29 PST 2015 suselsf28 : hist_test-histwin2 /remote/us01home33/croysdal 714 grep_hist_tmux_window echo
Thu Feb 5 15:11:49 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 712 grep_hist_tmux_window echo
Thu Feb 5 15:12:02 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 712 grep_hist_tmux_window echo

[716] /remote/us01home33/croysdal
28 us01home33/croysdal grep_hist_tmux_window echo
Thu Feb 5 15:09:51 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 709 echo "histwin1 cmd1"
Thu Feb 5 15:09:56 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 710 echo "histwin1 cmd2"
Thu Feb 5 15:11:49 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 712 grep_hist_tmux_window echo
Thu Feb 5 15:12:02 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 712 grep_hist_tmux_window echo
Thu Feb 5 15:14:38 PST 2015 suselsf28 : hist_test-histwin1 /remote/us01home33/croysdal 715 grep_hist_tmux_session echo
```

Useful functions / aliases

- zman Documentation for zsh commands
- snapshots Show snapshots for file(s)
- Man Open man page with gvim
- g Open file in existing gvim on page
- funlist List of functions

Limit `ls` to readable files/directories

```
> readable() { [[ -r $REPLY ]] }  
> alias lsr='ls *(+readable) '  
> lsr -ld
```

Bindings

- Defined in `~/.zsh/zshbindings`
- `bindkey` List all bindings

Emacs bindings

- Ctrl-A Jump to front of line
- Ctrl-E Jump to end of line
- Ctrl-B Backward character
- Ctrl-F Forward character
- Ctrl-D Delete char
- Ctrl-K Kill to end of line
- Ctrl-U Kill whole line

Emacs bindings

- Ctrl-X Ctrl-B Jump to matching bracket {}
- Ctrl-X Ctrl-F Vi 'f' command (find next char)
- Ctrl-X Ctrl-U Undo edit
- Ctrl-X ? Which command
- Ctrl-X Ctrl-N
 - Search in the history list for a line matching the current one and fetch the event following it.

Word bindings

- Word defined by:

```
export WORDCHARS='*?_[]~=&;!#$%^(){}.:'  
autoload -U select-word-style  
select-word-style normal
```

- <Esc> H Kill word backward (also Ctrl-W,<Esc>Backspace)
- <Esc> D Kill word forward
- <Esc> B Jump word backward
- <Esc> F Jump word forward
- <Esc> U Uppercase word forward
- <Esc> L Lowercase word forward

fzf -- fuzzy finder

- <https://github.com/junegunn/fzf> (Demo)
- Bindkeys
 - Ctrl-R Fuzzy history widget
 - Ctrl-T Ctrl-T File widget
 - Alt-C cd widget
- Commands (~/.zsh/fzf_funcs)
 - fs Switch tmux session
 - fe Edit file in gvim (uses gvim_tab)
 - fkill Kill process
 - z Shows fzf prompt when run with arguments
 - ficc Fuzzy search all ICC commands/vars

fzf -- fuzzy finder persistent history

- Easy access to persistent history across all shells
- Opens complete history with fzf prompt
- Commands :
 - fht full history from today (all shells)
 - fhw full history from week (all shells)
 - fhm full history from month (all shells)
 - fhf full history (all shells)

fzf -- fuzzy finder tmux

- Commands :

fw Find tmux window and switch to it

fs Find tmux session and switch to it

Vi bindings

- Ctrl-X Ctrl-V Switch to vi mode
- Ctrl-X Ctrl-E Edit command line in vim
- ***In VI mode***
- v Edit command line in vim
- bindkey -M vicmd List all vi bindings

History Bindings

- Ctrl-R Search history for pattern backwards
- Ctrl-S Search history for pattern forwards
- <Esc> . Last word from previous command line
Run multiple times for previous lines
- <Esc> m Previous word on history line
Run multiple times for previous words
- Arrows Go up/down history
- Ctrl-N/P Go up/down history
- Ctrl-X m Complete most recent file
 - Complete the name of the most recently modified file matching the pattern on the command line.

TMUX pane completion Binding

- <https://gist.github.com/blueyed/6856354>
- <http://blog.plenz.com/2012-01/zsh-complete-words-from-tmux-pane.html>
- **Ctrl-X t** to do a *prefix* completion: only words from the pane that share the same prefix will be presented
- **Ctrl-X Ctrl-X** to do a "find stuff like crazy" completion. If you see the output and just enter something from the middle of the word, it'll just as well complete.
 - For example, if you see 176.9.247.89 somewhere in the pane, try typing .9 and hitting Ctrl-X twice. It'll complete to that IP address.
-

Misc bindings

- F1 Show help
- Ctrl-T Show directory stack (dirs -v)

TODO

- setopt correctall
 - Corrects 2nd option to cp. Might disable

Zsh URLs

- <http://grml.org/zsh/zsh-lovers.html>
- <http://zshwiki.org/home/>
- <http://www.nparikh.org/unix/prompt.php>
- <http://explainshell.com/>
- <http://ivan.kanis.fr/zsh.pdf>
- <http://www.slideshare.net/jaguardesignstudio/why-zsh-is-cooler-than-your-shell-16194692>
- [Zsh glob](#)