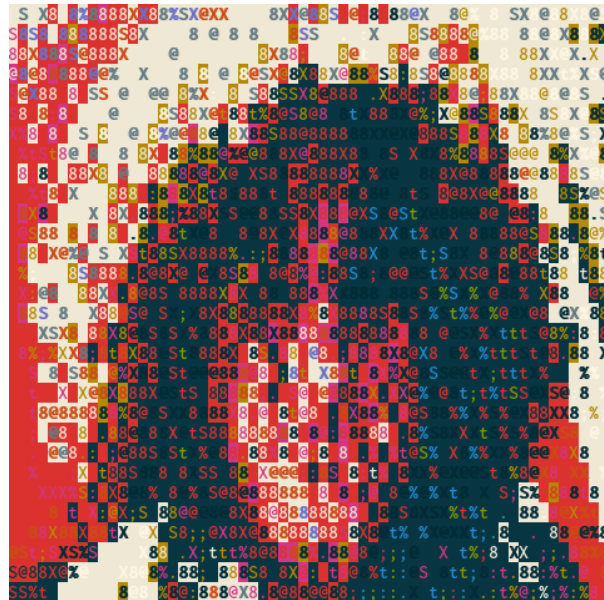


What the tmux?

Dotfiles Indy Meetup | 08.2020 | Clayton Parker

Who Am I?



@claytron on the internets and IRL

My dotfiles

What is it?

`tmux == Terminal MUltipleXer`

A better screen

tmux VS screen

Pros

- Client / server model
- Better keybinding support
- Multiple paste buffers
- More modern overall
- BSD licensed

Cons

- No serial / telnet terminal
- No support for older platforms and odd terminals

How it works?

Key Bindings

The prefix in tmux is `ctrl + b`

To avoid a conflict with screen's `ctrl + a`

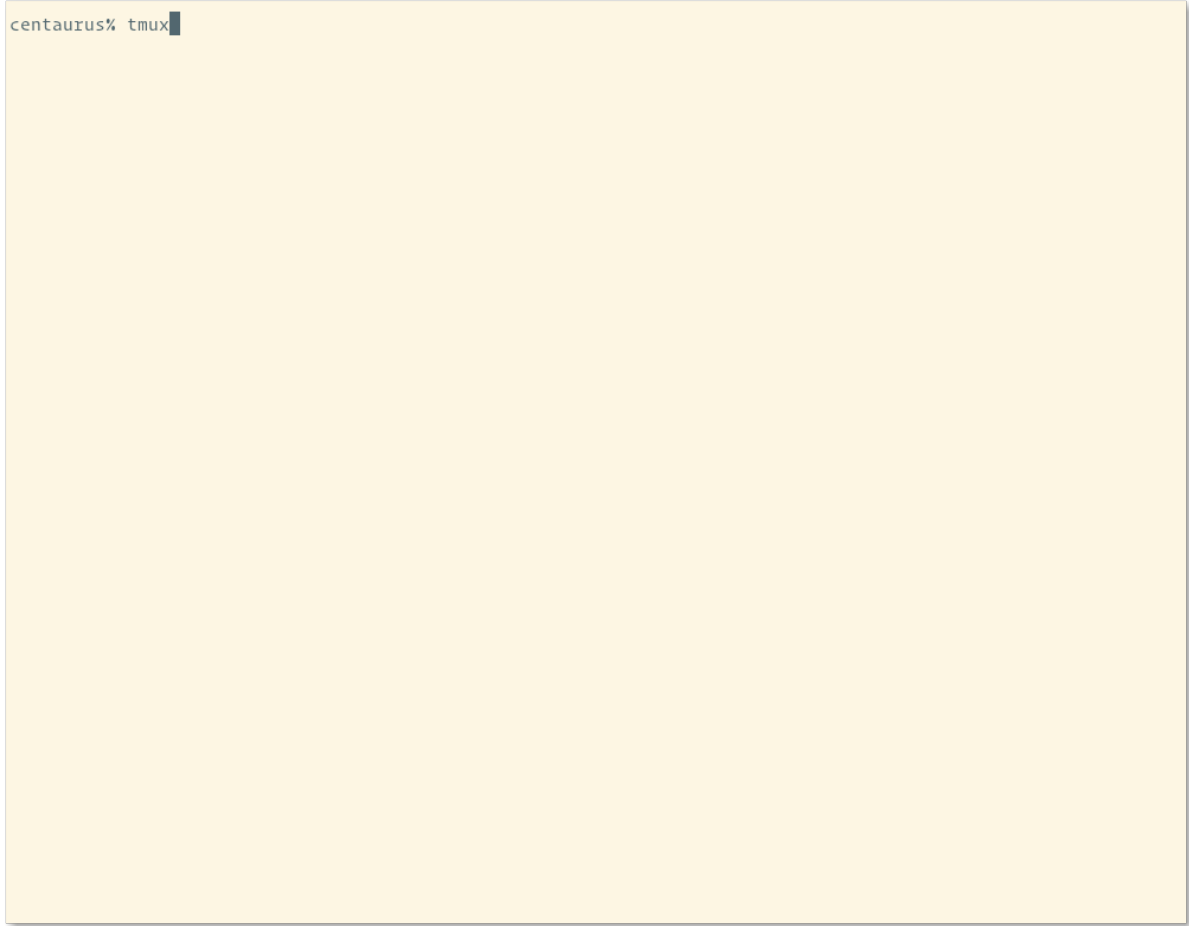
Remap it in your `tmux.conf` if needed

```
# Change the prefix and release ctrl + b
set -g prefix C-a
unbind C-b

# Ability to send ctrl + a to applications still
bind C-a send-prefix
```

Starting a new session

```
centaurus% tmux
```



Sessions



Sessions

The image shows a screenshot of a tmux terminal window. The title bar at the top is olive green and contains the text "(0) + 0: 1 windows (attached)". The main area of the terminal is a light yellow background. In the lower-left corner, there is a black rectangular box representing a terminal window. Inside this box, the text "0 (sort: index)" is on the first line, and "centaurus%" is on the second line with a black cursor. In the center of the yellow area, there is a small white box with a black border containing the text "0:zsh" in red. At the bottom of the terminal window, there is a dark green status bar. On the left side of the status bar, it says "[0] 0:[tmux]*". On the right side, it says "\"centaurus\" 09:38 20-Aug-20".

```
(0) + 0: 1 windows (attached)
```

```
0 (sort: index)
centaurus% █
```

```
0:zsh
```

```
[0] 0:[tmux]* "centaurus" 09:38 20-Aug-20
```

prefix + s

Detaching


centaurus% █

prefix + d

Starting another session

```
centaurus% tmux
```

Sessions

centaurus% 

```
[1] 0:zsh*
```

"centaurus" 11:29 20-Aug-20

Sessions

```
(0) + 0: 1 windows
(1) + 1: 1 windows (attached)
```



```
1 (sort: index)
centaurus% █
```

0:zsh

```
[1] 0:[tmux]* "centaurus" 09:38 20-Aug-20
```

prefix + s

Attaching

centaurus% █

Attaching

```
centaurus% tmux list-sessions
0: 1 windows (created Thu Aug 20 09:37:20 2020)
1: 2 windows (created Thu Aug 20 09:38:31 2020) (attached)
centaurus%
centaurus% tmux attach-session -t 0
```

Attaching

```
centaurus% █
```

```
[0] 0:zsh* "centaurus" 09:38 20-Aug-20
```


Windows and Panes

Windows and Panes



Split Vertically



prefix + %

Split Horizontally



prefix + "

Panes

```
(0) + 0: 1 windows
(1) - 1: 1 windows (attached)
(2)  └─> + 0: [tmux]*Z (3 panes)
```

0 (sort: index)

centaurus% █

centaurus% █

centaurus% █

0

1

2

```
[1] 0:[tmux]*Z "centaurus" 11:30 20-Aug-20
```

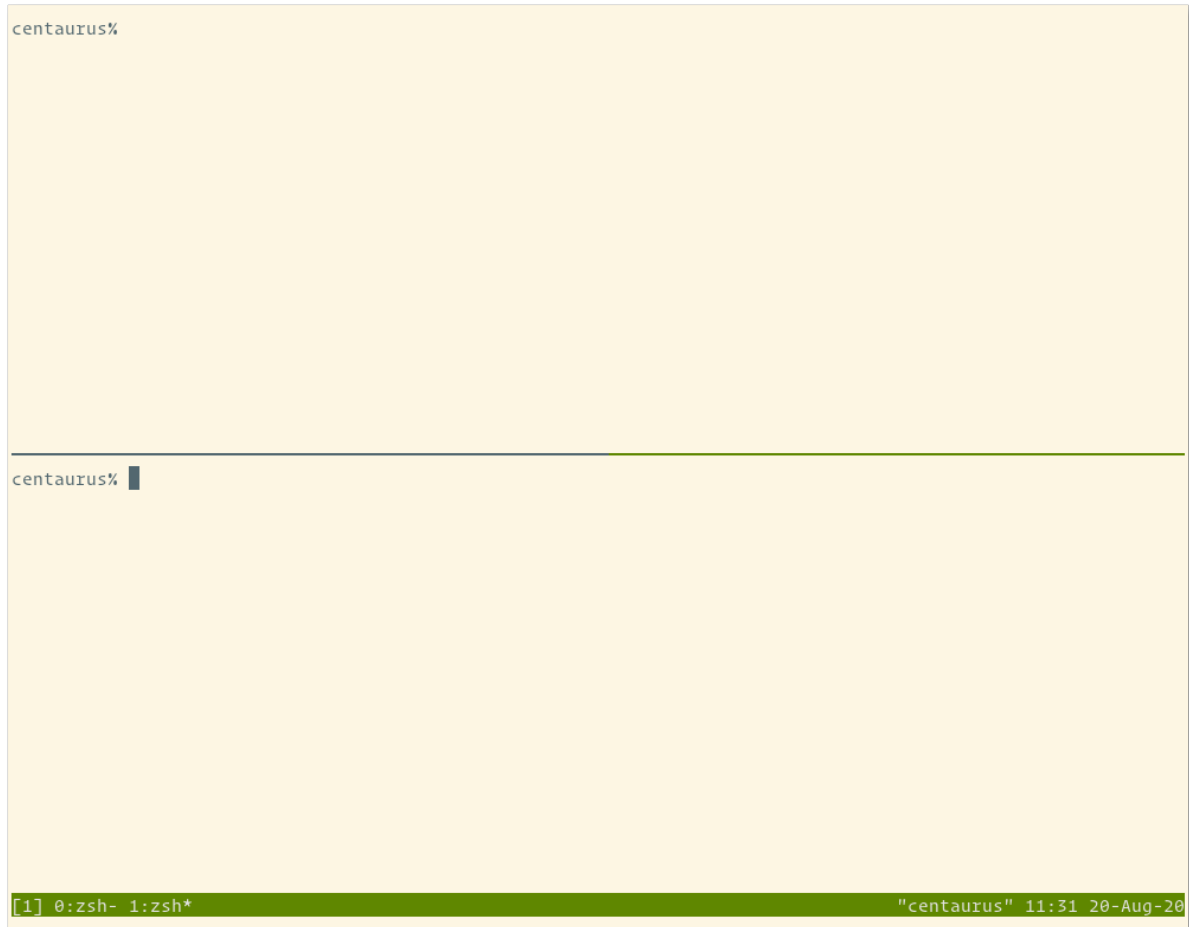
prefix + s

New Window



prefix + c

New Window



```
centaurus%  
  
centaurus%  
  
[1] 0:zsh- 1:zsh* "centaurus" 11:31 20-Aug-20
```

prefix + "

Session List

```
(0) + 0: 1 windows
(1) + 1: 2 windows (attached)
```

1 (sort: index)
centaurus% █
centaurus% █

0:zsh

1:[tmux]

```
[1] 0:zsh- 1:[tmux]*Z "centaurus" 11:33 20-Aug-20
```

prefix + s


```
(0) + 0: 1 windows
(1) - 1: 2 windows (attached)
(2) └─> + 0: zsh- (3 panes)
(3) └─> + 1: [tmux]*Z (2 panes)
```

```
1 (sort: index)─
```

```
centaurus% █
```

```
0:zsh
```

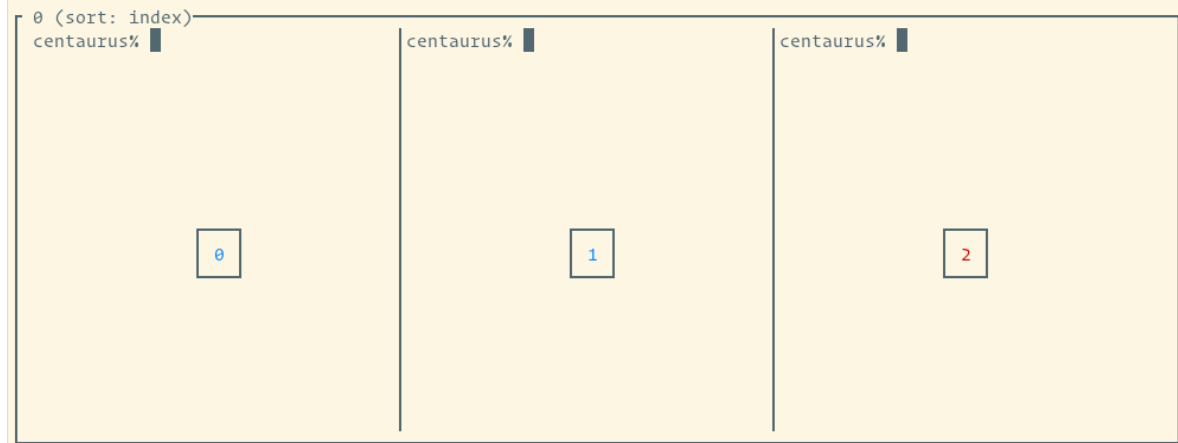
```
centaurus% █
```

```
1:[tmux]
```

```
[1] 0:zsh- 1:[tmux]*Z
```

```
"centaurus" 11:33 20-Aug-20
```

```
(0) + 0: 1 windows  
(1) - 1: 2 windows (attached)  
(2) └─> + 0: zsh- (3 panes)  
(3) └─> + 1: [tmux]*Z (2 panes)
```



[1] 0:zsh- 1:[tmux]*Z

"centaurus" 11:33 20-Aug-20

```
(0) + 0: 1 windows
(1) - 1: 2 windows (attached)
(2) -> - 0: zsh- (3 panes)
(3)   |> 0: zsh "centaurus"
(4)   |> 1: zsh "centaurus"
(5)   |> 2: zsh "centaurus"
(6) -> + 1: [tmux]*Z (2 panes)
```

```
0 (sort: index)
centaurus% █ centaurus% █ centaurus% █
```

0	1	2
---	---	---

```
[1] 0:zsh- 1:[tmux]*Z "centaurus" 11:33 20-Aug-20
```

```
(0) + 0: 1 windows
(1) - 1: 2 windows (attached)
(2) └─ - 0: zsh- (3 panes)
(3)   │   └─ 0: zsh "centaurus"
(4)   │   └─ 1: zsh "centaurus"
(5)   │   └─ 2: zsh "centaurus"
(6) └─ + 1: [tmux]*Z (2 panes)
```

1 (sort: index)─

centaurus% █

0

centaurus% █

1

[1] 0:zsh- 1:[tmux]*Z

"centaurus" 11:33 20-Aug-20

```
(0) + 0: 1 windows
(1) - 1: 2 windows (attached)
(2) └─ 0: zsh- (3 panes)
(3)   └─ 0: zsh "centaurus"
(4)   └─ 1: zsh "centaurus"
(5)   └─ 2: zsh "centaurus"
(6) └─ 1: [tmux]*Z (2 panes)
(7)   └─ 0: zsh "centaurus"
(8)   └─ 1: zsh "centaurus"
```

```
1 (sort: index)─
centaurus% █ centaurus% █
```

0

1

```
[1] 0:zsh- 1:[tmux]*Z "centaurus" 11:33 20-Aug-20
```

```
(0) + 0: 1 windows
(1) - 1: 2 windows (attached)
(2) └─ - 0: zsh- (3 panes)
(3)   │   └─ 0: zsh "centaurus"
(4)   │   └─ 1: zsh "centaurus"
(5)   │   └─ 2: zsh "centaurus"
(6) └─ - 1: [tmux]*Z (2 panes)
(7)   │   └─ 0: zsh "centaurus"
(8)   │   └─ 1: zsh "centaurus"
```

```
0 (sort: index)─
centaurus% █
```

0:zsh

```
[1] 0:zsh- 1:[tmux]*Z
```

```
"centaurus" 11:33 20-Aug-20
```

```
(0) - 0: 1 windows
(1)   ↳ 0: zsh* (1 panes) "centaurus"
(2) - 1: 2 windows (attached)
(3)   ↳ - 0: zsh- (3 panes)
(4)     ↳ 0: zsh "centaurus"
(5)     ↳ 1: zsh "centaurus"
(6)     ↳ 2: zsh "centaurus"
(7)   ↳ - 1: [tmux]*Z (2 panes)
(8)     ↳ 0: zsh "centaurus"
(9)     ↳ 1: zsh "centaurus"

0 (sort: index)
centaurus% █

0:zsh

[1] 0:zsh- 1:[tmux]*Z "centaurus" 11:33 20-Aug-20
```

Pane Selected



Naming

Sessions and Windows can have names

Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj    move down to start of next fold
# zk    move up to end of previous fold

# General tmux settings                                {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix
█

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@
```

[0] 0:vim*

"centaurus" 14:18 20-Aug-20

Naming

```
print('hello')
```

~~~~~

```
centaurus% tail -f some.log
```

```
centaurus% python3 hello.py
hello
centaurus% █
```

```
[1] 0:zsh* 1:python3-
```

"centaurus" 14:18 20-Aug-20

# Naming

```
print('hello')
```

~~~~~

```
centaurus% tail -f some.log
```

```
centaurus% python3 hello.py
hello
centaurus%
```

```
(rename-session) work
```

prefix + \$

Naming

```
print('hello')
```

~~~~~

```
centaurus% tail -f some.log
```

```
centaurus% python3 hello.py
```

hello

centaurus%

```
[work] 0:zsh* 1:python3-
```

"centaurus" 14:19 20-Aug-20

# Naming

```
print('hello')
```

~~~~~

```
centaurus% tail -f some.log
```

```
centaurus% python3 hello.py
hello
centaurus%
```

```
(rename-window) hello world
```

prefix + ,

Naming

```
print('hello')
```

~~~~~

```
centaurus% tail -f some.log
```

```
centaurus% python3 hello.py
```

hello

centaurus%

```
[work] 0:hello world* 1:python3-
```

"centaurus" 14:20 20-Aug-20

# Naming

```
~  
~  
~  
~
```

```
>>> help(print)
```

```
Help on built-in function print in module builtins:
```

```
print(...)
```

```
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)
```

```
    Prints the values to a stream, or to sys.stdout by default.
```

```
    Optional keyword arguments:
```

```
    file:  a file-like object (stream); defaults to the current sys.stdout.
```

```
    sep:   string inserted between values, default a space.
```

```
    end:   string appended after the last value, default a newline.
```

```
    flush: whether to forcibly flush the stream.
```

```
lines 1-11/11 (END)
```

```
centaurus%
```

```
(rename-window) docs
```

prefix + ,



# Naming

```
~  
~  
~  
~
```

```
>>> help(print)
```

```
Help on built-in function print in module builtins:
```

```
print(...)
```

```
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)
```

```
    Prints the values to a stream, or to sys.stdout by default.
```

```
    Optional keyword arguments:
```

```
    file:  a file-like object (stream); defaults to the current sys.stdout.
```

```
    sep:   string inserted between values, default a space.
```

```
    end:   string appended after the last value, default a newline.
```

```
    flush: whether to forcibly flush the stream.
```

```
lines 1-11/11 (END)
```

```
centaurus%
```

# Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj    move down to start of next fold
# zk    move up to end of previous fold

# General tmux settings                                     {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix
█

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@
```

[0] 0:vim\*

"centaurus" 14:18 20-Aug-20

# Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj     move down to start of next fold
# zk     move up to end of previous fold

# General tmux settings {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@

(rename-session) personal
```

prefix + \$

# Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj    move down to start of next fold
# zk    move up to end of previous fold

# General tmux settings {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix
█

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@

[personal]0:vim* "centaurus" 14:21 20-Aug-20
```

# Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj    move down to start of next fold
# zk    move up to end of previous fold

# General tmux settings {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@

(rename-window) dotfiles
```

prefix + ,

# Naming

```
# My tmux config uses folds in vim.
#
# zR    open all folds
# zM    close all folds
# za    toggle fold at cursor position
# zj    move down to start of next fold
# zk    move up to end of previous fold

# General tmux settings {{{1
# -----

# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix
█

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with applications like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
@

[personal]0:dotfiles* "centaurus" 14:22 20-Aug-20
```

# Naming

```
(0) - personal: 1 windows (attached)
(1)   ↳ 0: dotfiles* (1 panes) "centaurus"
(2) - work: 2 windows
(3)   ↳ + 0: hello world- (3 panes)
(4)   ↳ + 1: docs* (2 panes)
```

```
personal (sort: index)
# set the command prefix to match screen
set -g prefix C-a
unbind C-b
# Allow for C-a C-a to send it to the application
bind C-a send-prefix

# Set the proper terminal type
set -g default-terminal "tmux"

# Set the delay so that it doesn't interfere with 0:dotfiles ns like
# vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1
```

```
[personal]0:dotfiles* "centaurus" 14:22 20-Aug-20
```

prefix + s

# Naming

```
(0) - personal: 1 windows (attached)
(1)   ↳ 0: dotfiles* (1 panes) "centaurus"
(2) - work: 2 windows
(3)   ↳ + 0: hello world- (3 panes)
(4)   ↳ + 1: docs* (2 panes)
```

```
work (sort: index)
centaurus% python3 hello.py
hello
centaurus% █
```

0:hello world

1:docs

nes 1-11/11 (END)

[personal]0:dotfiles\*

"centaurus" 14:22 20-Aug-20



# Key Bindings

Vi and Emacs layouts for copy mode

Defaults are vague and hard to remember

Remap things to suit your style (read the Brian Hogan book)

# Pane Movement

By default movement between panes is

`prefix + <arrows>`

# Pane Movement

Set up vi style movement

```
bind C-h select-pane -L  
bind C-j select-pane -D  
bind C-k select-pane -U  
bind C-l select-pane -R
```

# Pane Resizing

```
bind -r H resize-pane -L 5  
bind -r J resize-pane -D 5  
bind -r K resize-pane -U 5  
unbind L  
bind -r L resize-pane -R 5
```

# Splitting

```
unbind '''  
unbind %  
bind | split-window -h -c "#{pane_current_path}"  
bind - split-window -v -c "#{pane_current_path}"  
bind c new-window -c "#{pane_current_path}"
```

# Modality

Like vi, tmux has different modes

# Command Mode

centaurus%

:list-sessions

prefix + :

# Command Mode



A screenshot of a terminal window with a light yellow background. The terminal shows the output of the `tmux` command in command mode. The output lists the current session and its windows. The status bar at the bottom is green and shows the session name, window number, and the host and time.

```
[2] 1 windows (created Thu Aug 20 16:14:34 2020) (attached) [0/0]
personal: 1 windows (created Thu Aug 20 12:12:29 2020) (attached)
work: 2 windows (created Thu Aug 20 12:14:27 2020)

[2] 0:[tmux]* "centaurus" 16:46 20-Aug-20
```



# Copy Mode

```
# pane window client [0/118]

# Put switch client back into place (was L)
unbind ""
bind -r "" switch-client -l

# Switch to last window
unbind ';'
bind -r ';' last-window

# Re-define switch to last pane, so this is easier to remember
bind -r l last-pane

# Appearance Settings {{{1
# -----

# Show notices when there is activity on another window
setw -g monitor-activity on
set -g visual-activity off

# Theme based on tmuxline
source-file ~/.tmux/tmuxline.tmux

# Status Line {{{1
# -----

# Force utf-8
set -gq status-utf8 on

# Localized settings for an OS {{{1
# -----

source-file ~/.tmux.local

# turn on folds
# vim: fdm=marker
centaurus%
[personal]0:dotfiles* "centaurus" 17:46 20-Aug-20
```

prefix + [

# Copy Mode

```
# pane window client [e/118]

# Put switch client back into place (was L)
unbind ""
bind -r "" switch-client -l

# Switch to last window
unbind ';'
bind -r ';' last-window

# Re-define switch to last pane, so this is easier to remember
bind -r l last-pane

# Appearance Settings {{{1
# -----

# Show notices when there is activity on another window
setw -g monitor-activity on
set -g visual-activity off

# Theme based on tmuxline
source-file ~/.tmux/tmuxline.tmux

# Status Line {{{1
# -----

# Force utf-8
set -gq status-utf8 on

# Localized settings for an OS {{{1
# -----

source-file ~/.tmux.local

# turn on folds
# vim: fdm=marker
centaurus%
(search up) v1
```

# Copy Mode

```
# pane window client (1/22 results) [0/118]

# Put switch client back into place (was L)
unbind ""
bind -r "" switch-client -l

# Switch to last window
unbind ';'
bind -r ';' last-window

# Re-define swtich to last pane, so this is easier to remember
bind -r l last-pane

# Appearance Settings {{{1
# -----

# Show notices when there is activity on another window
setw -g monitor-activity on
set -g visual-activity off

# Theme based on tmuxline
source-file ~/.tmux/tmuxline.tmux

# Status Line {{{1
# -----

# Force utf-8
set -gq status-utf8 on

# Localized settings for an OS {{{1
# -----

source-file ~/.tmux.local

# turn on folds
# im: fdm=marker
centaurus%
[personal]0:dotfiles* "centaurus" 17:24 20-Aug-20
```

# Copy Mode

```
set -sg escape-time 0 [89/118]

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# use vi key bindings
setw -g mode-keys vi

# turn on mouse mode
setw -g mouse on
# Don't mark the window on right click
unbind -T root MouseDown3Pane
bind -n WheelUpPane if-shell -F -t = "#{mouse_any_flag}" "send-keys -M" "if -Ft= '#{pane_in_mode}' 'send-keys -M' 'copy-mode -e'"
bind -n WheelDownPane select-pane -t= \; send-keys -M
# Default is to copy and cancel selection, this copies to the clipboard and leaves the selection
bind-key -T copy-mode-vi MouseDragEnd1Pane send-keys -X copy-pipe "pbcopy" \; display-message 'text copied to clipboard!'

# copy pasta
bind-key -T copy-mode-vi Enter send-keys -X copy-pipe "pbcopy" \; display-message 'Text copied to clipboard!'
bind-key -T copy-mode-vi y send-keys -X copy-pipe "pbcopy" \; display-message 'Text copied to clipboard!'
# vi like copy mode
# from https://twitter.com/nickmorrott/status/928343075646210050?s=09
bind -T copy-mode-vi v send-keys -X begin-selection
bind -T copy-mode-vi V send-keys -X select-line
bind -T copy-mode-vi C-v send-keys -X rectangle-toggle \; send-keys -X begin-selection
bind -T copy-mode-vi Escape send-keys -X cancel

# set the scrollbar
set -g history-limit 200000

# force tmux to use utf-8
setw -gq utf8 on

# Custom key bindings {{{1
[personal]0:dotfiles* "centaurus" 17:25 20-Aug-20
```

# Copy Mode

```
# pane window client

# Put switch client back into place (was L)
unbind ""
bind -r "" switch-client -l

# Switch to last window
unbind ','
bind -r ',' last-window

# Re-define switch to last pane, so this is easier to remember
bind -r l last-pane

# Appearance Settings {{{1
# -----

# Show notices when there is activity on another window
setw -g monitor-activity on
set -g visual-activity off

# Theme based on tmuxline
source-file ~/.tmux/tmuxline.tmux

# Status Line {{{1
# -----

# Force utf-8
set -gq status-utf8 on

# Localized settings for an OS {{{1
# -----

source-file ~/.tmux.local

# turn on folds
# vim: fdm=marker
centaurus%
:ls-buffers
```

# Copy Mode

```
buffer0: 90 bytes: "# Make window and pane indexes start with 1\012set -g base-index 1\012setw -g pane-ba[0/0]  
dex 1\012"
```

```
[personal]0:dotfiles*
```

```
"centaurus" 17:25 20-Aug-20
```

# Enhance Copy Mode

```
bind-key -T copy-mode-vi Enter send-keys -X copy-pipe "pbcopy" \; display-message  
bind-key -T copy-mode-vi y send-keys -X copy-pipe "pbcopy" \; display-message 'Tex  
  
# vi-ish bindings  
bind -T copy-mode-vi v send-keys -X begin-selection  
bind -T copy-mode-vi V send-keys -X select-line  
bind -T copy-mode-vi C-v send-keys -X rectangle-toggle \; send-keys -X begin-selec  
bind -T copy-mode-vi Escape send-keys -X cancel
```

# Mouse Mode

```
# turn on mouse mode
setw -g mouse on

# Don't mark the window on right click
unbind -T root MouseDown3Pane
bind -n WheelUpPane if-shell -F -t = "#{mouse_any_flag}" "send-keys -M" "if -Ft= ':"
bind -n WheelDownPane select-pane -t= \; send-keys -M

# Default is to copy and cancel selection, this copys to the clipboard and leaves
bind-key -T copy-mode-vi MouseDragEnd1Pane send-keys -X copy-pipe "pbcopy" \; disp
```



# More Config Examples

```
# Set the scrollbar
set -g history-limit 200000

# Set the delay so that it doesn't interfere with applications like vim
set -sg escape-time 0

# Make window and pane indexes start with 1
set -g base-index 1
setw -g pane-base-index 1

# Create a new session
bind S new-session
```

# Community

Plugins

Themes

Status line helpers

# Status line

```
1 # My tmux config uses folds in vim.
2 #
3 # zR    open all folds
4 # zM    close all folds
5 # zq    toggle fold at cursor position
6 # zi    move down to start of next fold
7 # zk    move up to end of previous fold
8
9 # General tmux settings {{{1
10 # -----
11
12 # set the command prefix to match screen
13 set -g prefix C-a
14 unbind C-b
15 # Allow for C-a C-a to send it to the application
16 bind C-a send-prefix
NORMAL > SPELL [EN] > +0 ~0 -0 .tmux.conf tmux utf-8[unix] 1/145 |1

( My name is Zach Galifianakis, I hope I'm pronouncing that right. )
(
( - Zach Galifianakis
-----
      ^__^
      o  (xx)\_______
         (__)\       )\/\
            ||----w |
            ||     ||

[ 16:34 Thu, Aug 20 — /home/clayton
[ centaurus% — █

personal > 1* > dotfiles 2020-08-20 < 16:34 centaurus
```

# Managing Environments

Teamocil, Tmuxinator, [And many more...](#)

# Clients

Language libraries like python, ruby, etc.

# Pair Programming

Using tools like wemux to remote pair

# Alternatives

Tiling Window Mangers

Builitn to terminal (e.g. iTerm2, Kitty, etc)

# Links

- [My Dotfiles](#)
- [Awesome tmux](#)
- [wemux](#)
- [Teamocil](#)
- [Tmuxinator](#)

# Books

- Brian Hogan's [tmux 2 Productive Mouse-Free Development](#)
- [Tao of tmux](#) free online book

These books are both based on tmux 2, not the latest, but still very informative



**FIN**