Deric Pang

[About](#)
[Publications](#)
[Projects](#)
[Blog](#)

A Minimum Viable Tmux Config

May 27, 2020

TLDR; see my config on [GitHub](#).

When I first started using Tmux, I was frustrated with its unintuitive defaults and keybindings. Over time, I pinpointed the features that I used the most and made them easier to access through better keybindings. I also customized my color scheme and status bar and made several quality of life improvements.

When going through this walkthrough, it's best to focus on *what* I'm configuring rather than *how*. Create a small custom config instead of copying someone else's since understanding how to extend and customize your tools is the key to unlocking their full potential.

1 What can Tmux do?

The main pieces of functionality that you will use in Tmux are:

- Splitting Tmux "windows" into multiple "panes", each of which is a separate shell session.
- Creating and switching between Tmux "windows" (which are more like tabs), each of which contains a number of panes.
- Creating Tmux sessions which are each a collection of "windows" and "panes".
- Detaching/switch from a Tmux session so that you can later pick up where you left off with all your windows and panes as you last had them.

2 Better Keybindings

Let's set up better keybindings for the parts of Tmux that we most commonly use.

2.1 Tmux Prefix

The Tmux prefix is a key chord that is used before all Tmux commands. I've gotten used to using the default, which is `C-b` (control followed by "b" without letting go of control), but it's common to use `C-a` instead. If you want to remap the prefix to `C-a`, add this to your `.tmux.conf`:

```
bind-key C-a send-prefix
```

2.1 Panes

Tmux "panes" are each a separate shell session, and each window holds one or more panes. You can add panes to a window in a way that is similar to splitting in a text editor.

Let's set up these keybindings:

- Vertically split the window with `C-b |`
- Horizontally split the window with `C-b -`
- Navigate between panes with `C-b [h, j, k, l]`
    - These are Vim-like keybindings where `h` is left, `j` is down, `k` is up, and `l` is right.
- Resize panes with `C-b [H, J, K, L]` (can repeatedly hit [H, J, K, L])
    - Again, these are Vim-like keybindings where `H` expands left, `J` expands down, `K` expands up, and `L` expands right.

Add the following to your `.tmux.conf`:

```
# Split panes using | and -. Start the new pane at the path of the current pane.
unbind-key '"'
unbind-key %
bind-key | split-window -h -c "#{pane_current_path}"
bind-key - split-window -v -c "#{pane_current_path}"

# Vim-like pane navigation and resizing.
bind-key h select-pane -L
bind-key j select-pane -D
bind-key k select-pane -U
bind-key l select-pane -R
bind-key -r H resize-pane -L 5
bind-key -r J resize-pane -D 5
bind-key -r K resize-pane -U 5
bind-key -r L resize-pane -R 5
```

2.1 Windows

Tmux "windows" are like the tabs in a web browser. Each window contains a number of "panes", each of which is a separate shell session. The default keybindings for managing windows are actually quite good, and I'll list out the important ones here:

- Create a new window with `C-b c`.
- Go to a particular window with `C-b [0-9]`.
- Go to the next window with `C-b n`.
- Go to the previous window with `C-b p`.
- Kill the current window with `C-b &`.
- Kill a particular window with `C-b :kill-window -t [0-9]`.

2.1 Sessions

Tmux "sessions" are collections of "windows" and "panes". I often have one session for each project I'm working on.

Let's set up the following keybindings:

- Create a new session with `C-b C-c`.
- Search for a session by name with `C-b C-f`.

Add the following to your `.tmux.conf`:

```
# Create new session.
bind-key C-c new-session -c "~"

# Search for a session.
bind-key C-f command-prompt -p find-session 'switch-client -t %%'
```

3 Status Bar & Color Scheme

I really like the Gruvbox[1] Vim color scheme, so I use a Gruvbox-like color scheme in Tmux heavily inspired by egel/tmux-gruvbox.

Add the following to your `.tmux.conf`:

```
# COLORSCHEME: gruvbox dark
set-option -g status "on"

set-option -g status-style bg=colour237,fg=colour223 # bg=bg1, fg=fg1

set-window-option -g window-status-activity-style bold,underscore

set-option -g pane-active-border-style fg=colour250 #fg2
set-option -g pane-border-style fg=colour237 #bg1
```

```
set-option -g pane-border-style fg=colour237 #bg1

set-option -g message-style bg=colour239,fg=colour223 # bg=bg2, fg=fg1

set-option -g message-command-style bg=colour239,fg=colour223 # bg=fg3, fg=bg1

set-option -g display-panes-active-colour colour250 #fg2
set-option -g display-panes-colour colour237 #bg1

set-option -g status-justify "left"
set-option -g status-left-style none
set-option -g status-left-length "80"
set-option -g status-right-style none
set-option -g status-right-length "80"
set-window-option -g window-status-separator ""

set-option -g status-left "#[fg=colour248, bg=colour241] #S #[fg=colour241, bg=colour237, nobold
set-option -g status-right "#[fg=colour248, bg=colour237, nobold, noitalics, nounderscore]#{?cli

set-window-option -g window-status-current-format " #[fg=colour239, bg=colour214] #I |#[fg=colou

set-window-option -g window-status-format " #[fg=#{?window_bell_flag,colour235,colour223},bg=#{?
```

If you're interested in a version of the status bar settings that use Powerline[2] symbols, check out my Tmux config on GitHub.

4 Quality of Life Improvements

There's a few things in my config that make working with Tmux easier:

```
# Reload tmux config with <prefix>-r.
bind-key r source-file ~/.tmux.conf \; display '~/.tmux.conf sourced'

# Set scroll history to 100,000 lines.
set-option -g history-limit 100000

# Lower the time it takes to register ESC.
set -s escape-time 0

# Mouse mode on.
set -g mouse on

# When scrolling with mouse wheel, reduce number of scrolled rows per tick to 1.
bind-key -T copy-mode-vi WheelUpPane select-pane \; send-keys -X -N 1 scroll-up
bind-key -T copy-mode-vi WheelDownPane select-pane \; send-keys -X -N 1 scroll-down

# Use Vim keybindings in copy mode.
set-window-option -g mode-keys vi
```

```
# Renumber windows when a window is closed.
set -g renumber-windows on

# Turn on activity monitors.
set -g monitor-activity on
set -g visual-activity off
```

5 Conclusion

That basically covers my entire Tmux config, which you can check out on GitHub. I hope this walkthrough helps you get started with Tmux!

---

1. https://github.com/morhetz/gruvbox↩
2. https://github.com/powerline/fonts↩