

Extracted from:

tmux 2

Productive Mouse-Free Development

This PDF file contains pages extracted from *tmux 2*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

Copyright © 2016 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The Pragmatic Bookshelf

Raleigh, North Carolina

tmux 2

Productive
Mouse-Free
Development



Brian P. Hogan

Edited by Susannah Davidson Pfalzer

tmux 2

Productive Mouse-Free Development

Brian P. Hogan

The Pragmatic Bookshelf

Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic books, screencasts, and audio books can help you and your team create better software and have more fun. Visit us at <https://pragprog.com>.

The team that produced this book includes:

Susannah Davidson Pfalzer (editor)

Nicole Abramowitz (copyedit)

Gilson Graphics (layout)

Janet Furlow (producer)

For sales, volume licensing, and support, please contact support@pragprog.com.

For international rights, please contact rights@pragprog.com.

Copyright © 2016 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.

ISBN-13: 978-1-68050-221-3

Encoded using the finest acid-free high-entropy binary digits.

Book version: P1.0—November 2016

Customizing the Status Line's Content

The tmux status line can display nearly any information we want. We can use some predefined components or create our own by executing shell commands.

The status line consists of three components: a left panel, the window list, and a right panel. By default, it looks like this:

```
[development] 0:bash*                               "example.local" 13:37 31-Oct-16
```

On the left side, we have the name of the tmux session followed by the list of windows. The list of windows shows the numerical index of the current window and its name. On the right side, we have the hostname of our server followed by the date and time. Let's customize the content of our status line.

Configuring Status Line Items

You can change the content in the left or right panels of the status bar using a combination of text and variables. The following table shows the possible variables we can use in our status line.

| Variable | Description |
|-------------------|--|
| #H | Hostname of local host |
| #h | Hostname of local host without the domain name |
| #F | Current window flag |
| #I | Current window index |
| #P | Current pane index |
| #S | Current session name |
| #T | Current window title |
| #W | Current window name |
| ## | A literal # |
| \$(shell-command) | First line of the shell command's output |
| #[attributes] | Color or attribute change |

Table 1—Status Line Variables

For example, if you wanted to show just the name of the current tmux session on the left, you'd use the set-option -g status-left option with the #S value, like this:

```
set -g status-left "#S"
```

But you can also make it stand out more by using an attribute to set the foreground color, like this:

```
set -g status-left "[fg=green]#S"
```

You can add as many attributes and items to the status line as you want. To demonstrate, let's alter the left side of the status line so it shows the session name in green, the current window number in yellow, and the current pane in cyan. Add this line to your configuration file:

```
set -g status-left "[fg=green]#S #[fg=yellow]#I #[fg=cyan]#P"
```

You can add any arbitrary text into the status line, too. Let's add text to make the session, window, and pane more noticeable, like this:

```
config/tmux.conf
# Status line left side to show Session:window:pane
set -g status-left-length 40
set -g status-left "[fg=green]Session: #S #[fg=yellow]#I #[fg=cyan]#P"
```

We set the `status-left-length` option because the output we've specified is too long for the default length, so we have to make that region wider.

You can configure the right side of the status line too. Add the current date and time, like this:

```
config/tmux.conf
# Status line right side - 31-Oct 13:37
set -g status-right "[fg=cyan]%d %b %R"
```

This formats the date as "31-Oct 13:37," but you can format it however you'd like, using the standard `strftime()` time formatting mechanism used in many programming languages.³ Your status line should now look like this:

```
Session: 0 2 11:~ - 2:~* 31 Oct 04:25
```

You can take things a step further by incorporating shell commands into the mix by using the `$(shell-command)` variable to return the result of any external command-line program into the status line. We'll go into this in detail in [Adding Battery Life to the Status Line, on page ?](#).

Keeping Status Line Info Up to Date

We've added the current time and some other dynamic information to our status line, but we need to tell tmux how often to refresh that information periodically. By default, tmux refreshes the status line every 15 seconds. We

3. See <http://www.foragoodstrftime.com/> for a handy tool to help you find the perfect time format.

can specify exactly how quickly tmux refreshes its status line with set-option -g status-interval followed by the refresh interval in seconds, like this:

```
# Update the status line every sixty seconds
set -g status-interval 60
```

This would refresh the status line every 60 seconds. Keep in mind that if you're firing off shell commands, those will be executed once per interval, so be careful not to load too many resource-intensive scripts.

Centering the Window List

We can also control the placement of the window list. By default, the window list is left-aligned, but we can center the window list in between the left and right status areas with a single configuration change:

```
config/tmux.conf
# Center the window list in the status line
set -g status-justify centre
```

With this in place, the window list appears centered:

```
Session: 0 2 1          1: bash- 2: bash*          31 Oct 04:26
```

As you create new windows, the window list will shift accordingly, staying in the center of the status line.

Identifying Activity in Other Windows

When you're working with more than one window, you'll want to be notified when something happens in one of the other windows in your session so you can react to it. You can do that by adding a visual notification, like this:

```
config/tmux.conf
# enable activity alerts
setw -g monitor-activity on
set -g visual-activity on
```

The monitor-activity on command highlights the window name in the status line when there's activity in that window. The visual-activity on line tells tmux to show a message in the status line as well.

Now when one of the other windows has some activity, it'll stand out with a cyan background, like the "top" window shown here:

```
Session: 0 2 1          1: bash  2: bash*  3: top#-          31 Oct 04:27
```

Once you switch to that window, the colors will revert back to normal. If you want to configure different colors, you can do so with `setw -g window-status-activity-style` and the colors of your choice.