

- **Ben Balter**

- [Posts](#)
- [Talks](#)
- [Press](#)
- [About](#)
- [Contact](#)

Attorney, open source developer, product manager

How I Atom (for prose).

Posted December 23, 2016 | [View revision history](#).

Inspired by @srobert's "[How I Atom](#)" [post](#), I thought I'd share a bit about my own computing environment. I've been using [Atom](#) as my day-to-day text editor (and probably the single most used app after my web browser) [since the day I joined GitHub](#), nearly four years ago, and about a year before [it was publicly released](#).

While at it's core, Atom is a simple, hackable text editor, what makes Atom so powerful is its extensibility and its ecosystem. There are [some 5,000 community-contributed packages](#) (and counting) that extend Atom's functionality and it's relatively easy to customize Atom's behavior using just HTML, CSS, and Javascript.

First and foremost, I'm a developer. While these days I may only occasionally write more than a few lines of code at any given time, over the years, I've grown accustomed to developer tools and developer workflows for all sorts of knowledge work, even if now, the majority of my time in Atom is spent authoring [Markdown](#) and prose. It's natural then, that I use a developer-centric text editor for the majority of my work, even if it's authoring English, and not say, Ruby.

Spending so much time in Atom, I've taken a great deal of care to shape every facet of my writing environment, and I'd like to share both some of the tweaks I've made, and how I manage them:

Atom packages I use

I highly recommend [taking a look at the packages @sroberts mentioned](#), if you haven't already, as I used most of them every day. Beyond that:

Quality of life

These packages shave seconds (sometimes minutes) off your day, and given their small size (and overhead), really set Atom apart from other text editors:

- [atom-alignment](#) - Have a bunch of -s, :s, or =s that your type-A mind can't stand to see vertically unaligned? Banish them with a single keystroke.
- [highlight-selected](#) - this one's hard to appreciate until you actually use it (or until it's missing). Double click a word to highlight all its uses in the current buffer. Useful for manual find-and-replace operations.
- [theme-switcher](#) - Switch between dark and light themes, helpful if you ever work outside.

Markdown

There's a number of Markdown-specific packages I use. Here's a few:

- [atomic-chrome](#) - One of my favorite Atom plugins, if you're in a textarea in Google Chrome, click the menu icon to open that textarea as an Atom editor, complete with all of Atom's bells and whistles.
- [markdown-lists](#) - If you're in the middle of a Markdown list and hit return, it inserts the bullet on the next line for you.
- [markdown-table-formatter](#) - If you create a Markdown table, on save, it aligns the columns such that the table is both human and machine readable.
- [wordcount](#) - Exactly what it sounds like, it adds a word (and character) count to the gutter. Useful for composing conference abstracts, Tweets, basically anything with a set limit.

Linters

If you're not familiar with the concept of a linter, it's basically an automated script that goes through the current file and highlights deviations from predefined stylistic rules, not too dissimilar to spelling or grammar mistakes on a standard word processor, and I can't recommend them enough.

linter-your-language-of-choice - Chances are, [there's a linter available for your language of choice](#), including prose. Whether you're learning the language or a veteran, the contextual hints it gives you will improve whatever it is you're writing.

Prose linters

There are a few prose-specific linters I credit for improving my day-to-day writing:

- [linter-write-good](#) - A grammar linter, Write Good checks all sorts of things, like cliches, passive voice, and unnecessarily wordy phrases.
- [linter-markdown](#) - This linter, based on [Remark JS](#) lints Markdown formatting, ensuring things like headings are surrounded by whitespace and bulleted lists are consistent.
- [linter-alex](#) - [Alex](#) checks your writing for biased or insensitive language.
- [linter-just-say-no](#) - Discourages you from using hedge words that weaken's the impact of your writing.

How I manage Atom

When I set up a new machine, among the first things I do is [symlink ~/.atom/config.cson and ~/.atom/keymap.cson to copies versioned as part of my dotfiles](#). Of course, if that's beyond your technical prowess (or willingness), or you need to keep secrets in your Atom config (like API keys), you can use something like [Mackup](#) to sync the config files via Dropbox. Since Atom stores its config as nearly-human-readable .cson files, it's easy to version settings and keep them in sync across machines. If you're really interested, since my dotfiles are open source, you can even [browse my Atom config](#).

You may also notice that I have a [packages.txt](#) file in my Atom config, with each plugin I want to use listed on its own line. While not a standard implementation, to be sure, Atom does come bundled with its own Atom Package Manager (APM), which can be fed a list of package files, via the `apm install --packages-file packages.txt` command, to install a list of desired packages. You can even create that list from your existing plugins [with this one-liner](#).

Working largely in prose, I can't imagine using a more traditional word processor over something that once customized, is more powerful and more purpose built. I can't count the number of times I've simply pasted in text I'm proofreading and implemented the automated suggestions (or wrote ad hoc rules to check for my own style nits).

That's how I Atom, but there are a bajillion packages (and workflows) out there. What packages do you use that you can't live without? How do you Atom?



Named one of the [top 25 most influential people in government and technology](#) and Fed50's Disruptor of the Year, described by the US Chief Technology Officer as one of "the baddest of the badass innovators," and winner of the Open Source People's Choice Award, Ben Balter is a Product Manager at [GitHub](#), the world's largest software development network, where he oversees the company's community and safety efforts. Previously, Ben served as GitHub's Government Evangelist, leading the efforts to encourage government at all levels to adopt open source philosophies for code, for data, and for policy development. [More about the author →](#)

[Edit](#)

This content is open source.
Please [help improve it](#).



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name



Vincent Voyer • a year ago

One package I love as for configuration saving: <https://atom.io/packages/sy...>

Thanks for the list, love atomic-chrome

^ | ▾ • Reply • Share ›

ALSO ON BEN BALTER

Copyright notices for open source projects

1 comment • 3 years ago



Michael Benford — Great post, Ben. I wonder if you could elaborate on how contributions from companies (people contributing while on the

Twelve things a product manager does

1 comment • 2 years ago



Benjamin Knott — Thanks for posting Ben. I'm learning that my title as a technical account manager to a federal agency is becoming more

Our code deserves better

1 comment • 3 years ago



Joe Flood — It's done this way because the federal government values process over result. If you're a contracting officer in government, the

If you liked it then you should have put a URL on it

1 comment • 3 years ago



Gklyne — "part of the secret sauce that makes RESTful interfaces possible." Much as I like the overall thrust of this post, I have to take

- [Books for geeks](#)
- [Fine Print](#)
-