A WP ENGINE PUBLICATION
SINCE 2013

"ALL THE WORD THAT'S FIT TO PRESS"

STEVIE AWARD WINNER

FEATURE COMMUNITY

HABITS, SKILLS, AND TOOLS OF 10 WORLD-CLASS WORDPRESS DEVELOPERS

NICK SCHÄFERHOFF OCTOBER 1, 2015

These days, with a little bit of effort, anyone can learn to build a WordPress site and extend it to fit their needs. The tools are readily available in the form of the WordPress platform itself as well as themes and plugin.

Many of us have probably thought about "going pro" at some point. To leave the realm of the WordPress aficionado, site builder, and hobbyist behind to write plugins, build themes, contribute to the WordPress core, be one of those people behind all the software us mortals use on a daily basis — in short, become a WordPress developer.

However, how do you get to that level? And what's even more interesting, how do you get to the top of that level?

To find out (or at least get an idea), I reached out to a number of well-known members of the WordPress community and asked them three questions:

1. In your personal opinion, which habits distinguish top-level developers? What do they do differently than others?
2. What habits and rituals do you personally adhere to that allow you to produce consistently good work?
3. What are your most important tools for WordPress development?

Thankfully, many of them were so kind to craft replies. So, if you are thinking of joining their ranks, the answers below should be able to point you in the right direction.

And now without further ado, the habits and tools of world-class WordPress developers (in no particular order).

Daniel Pataki: Branch Out, Be Humble, And Listen To Others

Daniel Pataki is a WordPress and PHP developer who likes to share what he learns during work with others. He is the editor of the WordPress section of Smashing Magazine and a contributor to the WPMU DEV blog among others. You can find a number of Daniel's plugins and themes in the WordPress repository and on GitHub.

How to become a top-level WordPress developer:

I think that what separates the truly top-tier developers is the same as in any profession: a general interest and awareness of what is happening outside their core *sphere of focus and the ability to see the world in something other than black and white.*

Seeing nuances in each situation makes good developers great. Knowing that icon fonts are truly amazing, but also thinking about the accessibility implications, knowing how breaking backwards compatibility in WordPress would be convenient, but would serve a few thousand developers and alienate millions. These are just some very simple examples of how top developers approach each question from multiple angles.

For development especially, after a certain level, it's not really about knowing more functions or how fast you can type. It's about learning how other systems than your main one (eg: WordPress) work, and slowly extending your reach to outlying territories (OOP, Gulp, Vagrant, App development, Accessibility, etc.)

Another common trait I've found in top-developers is the utter lack of arrogance and a much more humble outlook on life. The more you learn, the more you realize you don't know.

Personal work habits:

I'm probably an odd one out in this regard because I really do not like habits and rituals. I'm one of those people who will find something new, do it with full force for a month, then forget about it for years!

Trying to be as thorough as possible, being honest with myself and listening to what others have to say has helped me immensely throughout the years.

Perhaps one "habit" has been figuring out how to really listen to people. When your Mom says: "I loved the site you made, but the main button was difficult to find" you may dismiss it since the button is obviously right there! Realizing that one person's reaction (however faulty it may seem) probably reflects thousands upon thousands has helped me improve a lot.

Favorite tools:

I use Vagrant a lot for local development, WP-CLI for making setups and common tasks faster, Gulp as a build tool, Advanced Custom Fields for theme/plugin options (and a lot more), Github for managing/storing my projects and some others I'm sure I'm forgetting. The terminal in general was a hugely helpful tool once I got over the fear of using it.

Joe Casabona: Stay Thirsty, Keep Trying New Things

Next on the list is Joe Casabona. He works as a frontend developer at Crowdfavorite, an enterprise-level provider of WordPress services. Joe has also authored a book named Responsive Design with WordPress and a number of plugins. You can connect with him on Twitter or through his website.

How to become a top-level WordPress developer:

I would say the habits that distinguish top-level developers is their thirst to know more. This is a constantly changing industry and the folks who can stay on top of those changes are the ones who love to learn and enjoy the challenge of an ever-evolving industry.

I think a healthy dose of insecurity helps. Not so much that it's crippling, but enough to drive you to be better. I constantly feel like there's something else I should know or question if what I'm doing is the best way to do it. It prevents becoming complacent.

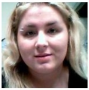
Personal work habits:

I read and try new things. I like to have a lot of side projects going to experiment. Even if some of them don't make it to launch, the skills I gain from working on them are invaluable. I also talk to other people in the field. Being in a silo is a sure way to let your work slide.

Favorite tools:

My most important tools are my editor (Atom), the Codex, and my coworkers. I work with a lot of smart people and I try to learn everything I can from them, whether it be through code reviews or conversations with them.

Nile Flores: Optimize Your Workflow And Adhere To Code Standards



One of the big household names in the WordPress community is that of Team Yoast and Nile Flores is part of it. Besides that, she is also a WordPress developer and blogger in her own right with an impressive portfolio. Every now and then Nile can also be found presenting at WordCamps and you can find her presentations on Slideshare.

How to become a top-level WordPress developer:

Top-level developers usually have put together a workflow that is more efficient and effective than those just starting out. For developers in the WordPress community, the top-level ones are utilizing tools for unit testing in different environments and always looking to improve the technology of their plugins and themes, while at the same time considering what their users really want. I see so many people complain that a plugin broke in some environment, only to ask the developer if they did any unit testing at all.

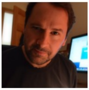
Personal work habits:

I usually try to adhere to best practices, especially code standards for WordPress. It's the only way to go. Ideally, it would be wonderful for my plugins and themes to be compatible with the thousands available, but the reality is, it won't. So, in first trying to adhere to WordPress code standards, and making sure they work with the core as well as default themes and plugins, it's definitely a good practice.

Favorite tools:

I love using PHPUnit and Ghostlab for my unit testing. I also check my work with Browserstack. For coding, I usually am using Notepad++. I also like using the developer tools in Firefox and Chrome. I also love to use Font Awesome.

Jeff Starr: Stick to Best Practices, Be Consistent, And Listen to Feedback



For those who don't know, Jeff is the guy behind Perishable Press, where he publishes WordPress tutorials and more. He also runs his own design business and has written several books such as The Tao of WordPress. Jeff's latest project is BBQ Pro, a plugin for blocking malicious queries on WordPress sites.

How to become a top-level WordPress developer:

I'm not sure about the habits of others, but I can tell you that adhering to WordPress best practices and guidelines is vital for successful development. That includes developing according to the WP API, keeping code updated, and testing thoroughly with the latest version of WordPress. It's also important to stay current with WordPress happenings and changes in the field of web design/development in general.

Personal work habits:

I think consistency begets consistency. So I try to maintain a consistent life routine and work schedule. While working, I try to be mindful of things like quality, security, and performance. Also, I think it's important to maintain good relationships with users and be responsive to user feedback. These things have served me well in my 10+ years working with WordPress. They also help when writing books and tutorials, which I do in addition to web development.

Favorite tools:

A code editor and an SFTP program, plus as many different browsers on as many different devices as possible. I try to keep my process simple and flexible, which helps to maximize the amount of time that I get to spend developing, writing, and creating awesome content.

Ben Lobaugh: See the Big Picture, Help Others, And Get Out of Your Head

When not found on his sailing boat, Ben is a lead developer at WebDevStudios. A former Code Wrangler at Automattic, he continues to actively contribute to the WordPress community by helping to organize WordCamp Seattle. Find him on Twitter or via is personal website.

How to become a top-level WordPress developer:

A typical answer here would be to 'love of developing things', however, I think it goes beyond that. Nobody gets into developing because they dislike it.

So what then sets the great developers apart?I believe that a great developer has two passions, the joy of engineering a project and a drive to help others succeed.

A top developer does not simply look at the part they are developing, they look at the whole and see the code as a form of poetry. Like a great poet, a great developer weaves together the grammar and syntax of their programming language to create something beautiful.

A top developer has a drive to help others succeed and grow beyond what they are currently capable of. By taking a system and breaking it into pieces that fit other developers skills and teaching them how to use the tools and the language to create more beautiful poetry of their own. This not only encourages more junior developers, it also strengthens the knowledge and skillset of the teacher.

One other thing I notice when I look at developers I admire is a hobby away from development. Something that allows you to switch off the programming part of your brain and experience life differently.

For me it is excelling at sailboat racing, for others is home brewing or hiking or music. As long as it is not utilizing your day to day development tools it allows your brain to relax and think in new patterns. It is great for the brain and makes you feel more refreshed next time you sit down to develop.

Personal work habits:

I have found that a major factor in work quality and quantity is the environment external to my computer. For me having a clean uncluttered workspace is a must or I will get distracted with stuff.

I often work at coffee shops to feel the presence of humans around me but may not want the noises and bits of conversation cutting into my thoughts. I use Senheisser earbuds both when working from coffee shops and working from home. They cut out all external sounds and have great clarity to boot.

Using multiple monitors also helps. I have found 3 to be a sweet spot but have had up to 6 with great success. When diving into actual development work I have a remote system that I connect to for all my coding. I have found this setup spectacular. I no longer have to rely on my laptop always being reliable as I can code from anything with a network connection, including my iPhone.

When starting a new project I do not look at a line item and dive in and start building it. Instead I read over the entire project and ask questions until the end product is firmly envisioned in my head. From there I sit back for a moment and visualize all the pieces and how they will fit together from the ground up to achieve the vision.

When you only look at the high level picture it is overwhelming to know where to start so from there I will break the project down into manageable chunks. Big enough to be interesting, but small enough to be accomplished quickly. Nothing is worse than spinning you wheels on a big chunk and feeling like nothing was accomplished during the day.

Favorite tools:

Really the community is my biggest tool. I would say that 95% of the time something I am trying to accomplish someone else in the community has done similarly and shared that info through the web. A quick web search usually brings me to the solution and if not Twitter has never failed me.

For my actual development I run a server at Digital Ocean that is fully configured with a suite of tools. Everything is pre-configured through scripts so getting running on a new project is quick and easy. The top 5 most used tools are: tmux, Vim, WP-CLI, Git, Grep

Glenn Ansley: Communicate Well, Deliver, And Seek Feedback

iThemes is another well-known name in the WordPress sphere and Glenn contributes to their high-quality plugins as a developer. He has also spoken at WordCamp Raleigh in recent years and you can find him talking about how to use iThemes Exchange on WordPress TV. Connect with Glenn on Twitter.

How to become a top-level WordPress developer:

Listen well: *You have to listen what your clients, customers, company, and coworkers value the most. If you're lucky, they use clear and descriptive words. More often, though, they speak to you through hesitation, frustration, incomplete ideas, praise, and even silence.*

Communicate Often: *Successful developers don't lock themselves in a cave for days on end. Give frequent updates to your coworkers, your supervisors, or your clients and customers. Ask coworkers or other devs for feedback when you hit a wall. Take a break and just enjoy the people around you. If you don't have people around, change locations for an afternoon. Humans need community to thrive and, contrary to popular belief, developers are humans too.*

Deliver results: *The result you need to deliver is different in each situation. It's status updates for your superiors or your clients. It's feature updates for your customers, code reviews for your coworkers, and tricks of the trade for junior devs. Know what you're responsible for and deliver it on time or your listening and communication skills are worthless.*

Personal work habits:

Manage the Chaos: *I close all my browser tabs and terminal sessions every night so that I start the morning with a clean slate. I only leave tabs and terminals sessions open during the day for what I'm currently working on at the moment. I use bookmarks and services like Trello to keep track of projects I don't currently have open. I limit interruptions by not leaving personal email and personal IM's open while I'm working. I'll check on those between switching projects throughout the day.*

Set goals and rewards: *Goals are for my work that needs to get done for the day. Specific goals that can be crossed off a list are best for my task-oriented personality. I allow myself short rewards in between goals. And by 'rewards', I mean things like using the restroom, getting a drink, taking a walk, and checking my email… I'm the face of adventure.*

Seek feedback: *Knowing that people appreciate what I'm doing and that I'm doing it to the standards they value really inspires me to work harder. I've been in situations before where I didn't seek feedback and wasn't offered any either. It's really easy to start doubting what you're doing in those instances… and that's very demotivating to me.*

Favorite tools:

I do all my work on a remote server via SSH, Vim, and heavy use of WP-CLI. This makes me very portable. My access to my work environment is not dependent on the computer I have in front of me, only an Internet connection and a shell of some sort.

The WordPress codebase, specifically, the latest version of trunk. It automatically updates multiple times a day so I'm always on the latest version in my development environment.

ack-grep is a command line tool that I use all-day, every day. It makes searching through core WP files or even my own plugin files extremely fast. This tool, combined with the excellent inline documentation in WordPress, means that I just lookup functions in the code before I go to the WP codex 100% of the time.

I also rely heavily on Git and several custom bash scripts to automate as much as I can.

Google and browser shortcuts. I have multiple 'Custom Search Engines' setup in Chrome to help me quickly find WordPress Trac tickets, codex pages, development sites, and other frequently visited sites.

Jazz music.

Carrie Dils: Never Stop Learning And Find Balance

Carrie has also recently become part of Crowdfavorite. Before that she was working as a freelancer and running her own WordPress and business blog. She is one of my favorite tutorial writers since I often get a chuckle out of her articles on Genesis. Be sure to check her weekly podcast on the business side of WordPress.

How to become a top-level WordPress developer:

They're continually in learning mode, actively improving their skills and challenging themselves in their work. They're also willing to submit themselves to the (constructive) criticism of others and actively seek out peer feedback.

Personal work habits:

Ha! I didn't know I had rituals, but I suppose I do. I could sum it up with I try to take care of myself. I keep the same general work hours daily. I'm not a fitness freak by any means, but at some point during the day I'll leave my desk for 90 minutes or so to either go on a bike ride, go to the gym, or do a little vigorous house work.

Taking care of myself mentally, physically, and spiritually means that I can be fully present and sharp when I sit down to work. Also, after some mishaps, I created a new rule for myself: No coding before coffee.

Favorite tools:

A good code editor (I'm using Sublime Text right now but considering a jump to PHPstorm), Git (I'm especially learning the importance of informative commit messages), and having a consistent workflow (everyone's looks different, of course, but consistency creates better efficiency and reduces the chance of missing something important).

George Stephanis: Adopt A Broad View, Be Humble, And Automate

George develops plugins for a living. However, not just any plugins. He is lead for Jetpack (one of **the** WordPress plugins) at Automattic, the company behind WordPress.com. You can also find him on Twitter and Github.

How to become a top-level WordPress developer:

Top-level developers have several unique traits occurring concurrently — the traits may all happen in some other combination in many or all developers, but managing most or all of them at the same time is what marks (for me, at least) an upper-echelon dev. These traits include:

- ***Consideration of the full life-cycle of a project.*** *Not merely how users will start to use your work, but also how they will stop using it and move on to something else.*
- ***Awareness of project extensibility.*** *Few projects perfectly suit all users. To cater to edge cases, make the code extensible, so as to support plugins that modify other plugins or the like.*
- ***Knowledge that you don't need to solve every problem.*** *One solution does not need to be all solutions.*
- ***Consistency with code style in a project.*** *This includes everything from whitespace to variable naming conventions, to extending Core styles for your own table instead of recreating from scratch.*
- ***Willingness to subordinate your own inclinations*** *for the good of the project / users / community / etc.*
- ***Humility sufficient to seek help with your shortcomings.*** *No one is an ace an all areas, so the ability to collaborate with others for a better end result is critical.*
- ***Wisdom enough to keep all the other considerations above in check****, and not become so obsessed with code style (for example) that it detrimentally affects backward compatibility.*

And probably some others that aren't immediately coming to mind. To distill it down to a bite-sized chunk, the advancement from Junior Dev to Senior Dev can be seen in a change of the predominant traits: Effort => Intelligence => Wisdom.

Personal work habits:

A friend once told me once "If a thing is worth doing more than once, it's worth automating, so you never have to do it manually again." — Now, obviously this isn't a universal, but when it's efficient and the added consistency can be of use, it's often worth switching things to automated tasks — like deploy processes, unit tests, *adding text domains — via Grunt or the like.*

Favorite tools:

My brain, and maybe a good single malt whisky. Everything else — IDE, Unit Tests, Code Linting — are nice to haves, but most can be done without. A well-trained mind capable of devising reliable solutions is really the most important tool one can have.

Mike Schinkel: Adhere to Best Practices And Know Your Tools

An entrepreneur and developer specializing in enterprise-level WordPress websites, Mike can be found through his website, Twitter and plugins in the WordPress directory. He also runs his own development company and graciously send in his replies on short notice.

How to become a top-level WordPress developer:

First, let me say that when you write "WordPress Developer" I think of someone who is experienced and comfortable coding PHP, is comfortable writing MySQL queries, and is experienced with at least jQuery development if not Javascript.

Too often people say "WordPress Developer" and mean "Someone who knows how to install WordPress; select, install and configure a theme and collect, install and configure plugins". That person is not a "developer" in my opinion, that person is a "Sitebuilder."

That said, I am not a habitual sort of person so I don't think in terms of "habits". But if you were to ask "best practices" for WordPress Developers then:

- *Uses version control, most likely Git*
- *Develops locally*
- *Employs (at least) a 3-stage deployment cycle: Test, Stage & Production*
- *Rigorous adherence to coding standards*
- *Continuous refactoring during development*
- *Focus on building reusable functionality*
- *Using a sold variable/function/class/method/property naming strategy with a high level of consistency across projects*
- *Separation of backend logic from front-end layout*
- *Use real OOP for modeling post types, terms and user roles, etc. and not just Classes-as-Namespaces (although that, too, is helpful.)*

Personal work habits:

Again, I don't think in terms of "habits." The answer to this question is really the same as the answer to the first question.

Favorite tools:

Hand's down: PHPstorm with XDebug. I've even started avoiding working with developers that do not use PHPstorm+XDebug because, in my experience, those who do not use PHPStorm+XDebug get stuck too easily, learn much slower (because using it is so educational related to WordPress core) and because they take much longer to finish a project.

Beyond that:

- *Navicat for MySQL*
- *iTerm (on Mac) for terminal*
- *Git for version control.*
- *Bitbucket for private Git hosting*
- *Github for public Git hosting*
- *VirtualHostX (on Mac) for local development configuration*
- *HTTP Client (on Mac) for inspecting HTTP requests and responses*
- *Pantheon for site hosting (when client's allow) because they provide a Git repo and Test/Stage Production URLs with workflow*
- *Jenkins for build server in front of Pantheon's Git repo*
- *JIRA for project task tracking*
- *Slack for team communications*

There are many other tools I use too, but the above are the standouts that I and my team would be hobbled without.

Lew Ayotte: Stay Curious, Focus, Plan, And Embrace Distractions

Last but not least we have Lew, another developer for iThemes (thanks to Cory Miller for making the connection to Lew an Glenn by the way!) and author of The Concise Guide to Securing WordPress and Repairing Hacks. You can check Lew's other plugins in the directory, learn more about him via his website and connect via Twitter.

How to become a top-level WordPress developer:

Can you call curiosity and experimentation a habit? I think those two things are probably the most effective habits that make someone a top-level developer. A person who wants to know how something works and is willing to try to figure it out.

These habits can be expressed in vastly different ways depending on the developer's personality but they all boil down to four things:

- *You need to have **a consistent source of knowledge** to rely on (books, forums, webchat, other friends who know more than you or are learning with you)*
- *You need to have the **time to learn** whatever you're trying to learn — sometimes this require **sacrifice**, but it's always worth it*
- *You need to be able to **learn from your failures and your successes** — I think it's easier to learn from your failures. But take a step back when you succeed and ask yourself, "Why did this work?" — many times, it'll help you come up with better solutions for the problem you're working through.*
- *You need to **challenge yourself** — If you're always doing only the things you're comfortable with, you'll eventually find yourself with a challenge that you won't be able to handle. When you push yourself, you not only learn more about what you're doing, you are better prepared for the future.*

I believe a top-level developer is someone who can put all these things together to build their skill set and set themselves apart from others. This doesn't happen over night, it can (and probably should) take years.

Personal work habits:

I work from home and already have a pretty ritualistic lifestyle and personality. I wake up around 7AM every morning, feed my dogs, feed my kids, feed my wife, feed myself, and start the day.

I find it hardest to produce consistently good work if my ritual is interrupted or if I know it's going to be interrupted. Knowing that I can only spend one hour on a task before an interruption (like a planned meeting) when I'm going to need three is going to mess up my productivity. So, I find it's important to plan around known interruptions.

Also, I find that I do my best work when I have classical music playing in the background. It provides me with noise to drown out any other distractions and allows me to better focus on the task at hand.

I stay away from social networks and I only check my email a few times a day, unless my brain needs a rest or distraction. Yes, sometimes you need a distraction! If I've been stuck on a problem for an hour and I've tried everything, I go take a shower or take a quick walk. It's amazing what your brain can do when you let it rest.

Favorite tools:

A version control system like Git or SVN is probably the most important tool you can have. Preferably Git, it makes life so much easier, if you haven't taken the time to learn it (or SVN), go do it right now.

After that a good file search tool, I prefer ack-grep on Linux because it has a lot of built-in functionality that is harder to produce easily with the default grep tool. This is really important not only for searching your own code but for searching WordPress core (or other plugins) for code that will help you. Combined with regular expressions and you can usually find everything you're looking for.

I also use Coda2 as my primary IDE and Vim as my editor of choice in Linux. These four tools are the most important tools I use for WordPress development.

Bonus Tips — Matt Mullenweg: Polyphasic Sleep, DVORAK, And Music

Matt Mullenweg, CEO of Automattic and one of the guys who started all the

WordPress craze was on Tim Ferris's podcast earlier this year. While the conversation revolved around a bunch of topics, Matt mentioned some habits and tools that have helped him become more productive over the years.

For the more experimental among you, there are polyphasic sleep schedules (which Matt gave up after getting a girlfriend). He also recommends learning the DVORAK keyboard layout instead of the standard QWERTY layout. Finally, he counts music as one his main productivity tools.

You can listen to the entire interview here and find the shownotes on the blog.

So, What Does It Take To Become World-Class?

Being at the top of the game in anything is not an easy feat. Being a WordPress developer is no exception. It takes hard work, dedication, and a number of other things.

Thanks to our 10 experts, we are now equipped with the knowledge on how to move ahead in this field. Since it was a lot of material, let's do a quick recap. Here are the main traits and tools our 10 developers mentioned:

Developer Traits

- **Curiosity** – Look beyond your own field, have a thirst for knowledge and challenge yourself
- **Humility** – Seek and accept feedback, try to help others
- **Communication** – Be available, listen properly, maintain good relationships
- **Vigilance** – Adhere to code standards and best practices, test thoroughly, see the full picture, be consistent, deliver on time
- **Balance** – Find a counterweight to work and development to keep sane

Developer Tools

- IDEs (Coda2)
- Code Editors (Atom, Notepad++, Vim, Sublime Text, PHPStorm)
- Command Line (WP-CLI)
- Version Control (Git, Github, Bitbucket, SVN)
- Automation (Gulp, Grunt)
- Testing (Browserstack, PHPUnit, Ghostlab)
- Resources (Codex, Codebase)
- Local Development (VirtualHostX)
- Debugging (XDebug)
- Project Management (JIRA, Slack, Trello)
- Codesearch (ack-grep)

Thanks so much to everyone who contributed! It's really nice to see how everyone took some time out of their busy schedule to share their insights with the rest of us. It's this kind of community thinking that makes WordPress so successful.

How do you keep yourself productive when building sites? Any tips for the rest of us? Tools, mindset? Please, share in the comments!

Nick Schäferhoff

Nick Schäferhoff is an entrepreneur, online marketer, and professional blogger from Germany. He found WordPress when he needed a website for his first business and instantly fell in love. When not building websites, creating content or helping his clients improve their online business, he can most often be found at the gym, the dojo or traveling the world with his wife. If you want to get in touch with him, you can do so via Twitter or through his website.

Related

100 WordPress Influencers to Follow in 2015

20 Of The Most Successful WordPress Businesses

17 Of The Best Code Editors For WordPress Developers And Users

40+ Awesome WordPress Development Tools

Powered by Contextly

There are 6 comments

kjbenk . · October 2, 2015 at 6:51 am

This is a great resource for developers who want to improve (so basically all developers). I think it would be nice to also have gotten Pippin's viewpoint on this, since I think he is a fantastic developer.

Reply

Nick · October 2, 2015 at 6:58 pm

Thanks man! Happy you like it. And I contacted Pippin as well (since I totally agree with you), unfortunately he had to decline because of time restrictions. Next time though.

Reply

masudicb · October 2, 2015 at 12:59 pm

This is really a great resource to learn about the world-class WordPress developers.

Reply

Nick · October 2, 2015 at 6:57 pm

Thanks Masud, I'm glad you like it. I also learned a ton talking to all of these talented people.

Reply

Nile Flores · October 4, 2015 at 2:58 am

Thank you for reaching out. I actually had a few male developers that reached out to me and were like "Hey, I didn't know you used that! I'm geeking out on that too."

Reply

Nick · October 5, 2015 at 2:10 am

Haha, nice. Nerdy minds think alike I guess? Thanks again for contributing to the article Nile!

Reply

Your email address will not be published.
Required fields are marked *

Comment

Name *	Email *	Website

☐ Continue the conversation via email!

POST COMMENT

More News from Torque

‹ What Happens At WordCamp Vegas, Hopefully Won't Stay In Vegas

Free WordPress Themes: September 2015 Edition ›

TORQUE®

A WP Engine Publication

© 2016 WPEngine, Inc. All Rights Reserved.
WP ENGINE®, TORQUE®, EVERCACHE®, and the cog logo service marks are owned by WPEngine, Inc.