



THE WORDPRESS REST API

AN UNTECHNICAL GUIDE FOR
NON-DEVELOPERS

By Nick Schäferhoff

TORQUE

 **WPengine**[®]

INTRO

A buzzword is sweeping the WordPress nation that has had developers on their toes for months.

They are not alone. Even casual WordPress users have definitely noticed something in their peripheral vision.

I am, of course, talking about the WordPress REST API (some people also refer to it as the JSON REST API), an upcoming addition to WordPress core. Currently available in [plugin](#) form and originally slated for version 4.1, we can expect its two-phased integration for versions 4.4 and 4.5.

Josh Pollock has written several articles on the topic (found [here](#), [here](#), [here](#), [here](#), and [here](#)), as well as an entire ebook, which you can [download for free](#).

However, while WordPress developers seem generally excited about this new feature, the average WordPress user might not understand what all the fuss is about.

That's a bit regretful for something that is considered one of the most groundbreaking developments in recent WordPress history.

For you non-technical users (like me), let's have a look at what exactly the WordPress REST API is (and why you have to capitalize it), as well as what it means for the future of WordPress.

WHAT IS AN API?

Let's start with the basics and some terminology. After all, the name JSON REST API is completely made up of acronyms (hence the capitalization).

Let's start with the last one: API.

The three letters "API" stand for *Application Programming Interface*.

What is it for?

Basically, an API provides a way to establish a connection between different pieces of software. Think of it like a USB

port. USB ports allow you to connect many types of devices to your computer: printers, control devices, portable hard drives, phones, coffee cup warmers, you name it. Through the port, your computer and the device in question can interact with one another by sending data back and forth. An API is kind of like that, only between two applications instead of physical devices. It allows two programs to exchange data in clearly defined, limited ways in order to work together.

You use APIs all the time, whether you are aware of it or not. For example, if you use a third-party program to run your Twitter account, this is done via the Twitter API. Thanks to API technology, I can be on Twitter every day without actually going to the Twitter webpage.

APIs are also in use when you connect an email marketing service like [MailChimp to WordPress](#). Or, if you are one of the handful (err, millions) of people who play Candy Crush, games like this rely on the Facebook API for login, data sharing, and more.

In short, Application Programming Interfaces allow developers to use features, data, or content from other web applications or services for their own software in a secure way.

Capisce? Alright then, let's move on.

WHAT DO REST AND JSON STAND FOR?

REST derives its name from *Representational State Transfer*. It describes a particular style of building APIs that's designed to be lightweight, user-friendly, and suitable for high-volume services.

Google, Facebook, and Twitter all use the REST approach for many of their APIs. One reason for this is because REST is based on HTTP; you know, the protocol that powers almost all Internet connections in the world (together with its safer cousin HTTPS).

Because of their high compatibility with today's technology, APIs based on REST have become extremely popular around the web, not just for WordPress. Pretty much any program that connects to the Internet can use this type of API. Sounds

like a lot of opportunity for cross-program collaboration, doesn't it?

JSON, on the other hand, is short for *JavaScript Object Notation*. The name describes a data-exchange format based on the JavaScript programming language.

What's special about JSON is that it is both human and machine friendly. Developers can read and write it like a normal programming language and it is easily parsed and generated by computers.

However, one of its main advantages is that most programming languages already have encoders and decoders to convert their data structure to JSON and back. That means a JSON interface can act as an interpreter between two applications in vastly differing programming languages that otherwise would never be able to communicate with one another.

The JSON format has become somewhat of a universal connector on the Internet, and is the reason WordPress and other big services out there opt for it.

Additionally, in the WordPress sphere JSON also replaces the slightly outdated [XML-RPC](#) standard, which is much more complicated to use. In fact, a lot of the old standard's code is devoted to making WordPress understand its own format. Doesn't sound very efficient if you ask me.

So, to sum it up:

- An API is like a socket on a piece of software
- REST is the socket's design
- JSON is a cable with universal plugs on all ends

Easy, wasn't it?

THE JSON REST API AND WORDPRESS

Ok, so this API stuff sounds useful in general. But doesn't WordPress already have APIs?

Well, yes it does actually, just one: the [WordPress API](#). It is used constantly for WordPress plugins and works just fine for

WordPress-internal processes. However, when interacting with software outside of the platform, it is outdated and not very user-friendly.

The new [WP API](#) (as it is called by now) is all about **universality**. It was built to let WordPress easily interact with other websites and services on the Internet.

While that may sound a bit underwhelming, it is actually quite groundbreaking.

This universal API makes it possible to display and save content from other sites and applications, **regardless of whether or not they are using WordPress themselves!**

You read that right.

The API basically allows the WordPress platform to become a content management system for any application in any programming language.

Plus, the reverse is also true.

Anything that is present on your WordPress website can now also be made available to external websites and services, including:

- Posts
- Pages
- Custom Post Types
- Media
- Comments
- Taxonomies
- Users
- and more...

This works because everything is based on HTTP, which is universally available. The protocol allows sites to send, create, read, update, and delete requests back and forth between two parties (so-called CRUD requests).

In HTTP, the equivalents are POST, GET, PUT, and DELETE.

In addition to that, the API understands directory-like URL structures such as the ones used by WordPress and other content management systems.

Combined, this means if you want to call all posts from a certain user from your website to another site, you can by using the following directive:

```
GET http://yoursite.com/users/username
```

The same can be applied for updating single posts, deleting users, publishing pages, and much more.

In short, the API lets you do anything the WordPress admin area is capable of, only remotely, without manually logging into your site and including the non-WordPress content.

Mind = blown...right?

Cool, But is it Safe?

At this point, a question that I have seen arise in connection with the API is that of security (for good reason). Who wants to give remote control of their website to someone else?

The good news is that, naturally, the WordPress REST API has built-in safety measures. The magic word here is **authentication**. The interface comes with both cookie and [OAuth](#) authentication.

While cookie authentication works for plugins and themes, OAuth is used to authenticate desktop, mobile, and web clients. Both methods limit third-party actions on your site. As mentioned earlier, APIs are built to allow **limited** and **clearly-defined** data exchange between platforms, not total control of one by the other. Therefore, your data is completely safe. To learn more about the security of the WordPress REST API, see [here](#).

WHAT DOES THE WP API MEAN FOR WORDPRESS?

Sounds all good and dandy so far. However, isn't this just one of those things only developers get excited about? You know, changes in the code? How does this affect the average user's website? I mean, after all, there was an API before and I didn't care about that one either.

Well, the reason for all the buzz around the WordPress REST API is that it puts WordPress on the path to becoming a full-fledged application platform. The API provides a standardized way for other applications and environments to interact with the WordPress platform, opening it up to the rest of the Internet.

This is huge! And here's why.

The WordPress Backend Becomes Completely Optional

Remember how I said earlier that the API can do the same things as the WordPress admin area without taking the detour over the user interface? This means the WordPress backend is no longer needed.

Any developer who has ever had a grudge with it will be able to build an alternative WordPress admin panel on—and here is the kicker—**virtually any device or platform**.

Yep, that's right. The new API is so versatile that developers are no longer bound to familiar environments. If someone wants to build an admin area or blogging client that works with but is not based on WordPress, they now can.

This is good news for users who might struggle with the WordPress back-end. The new API could bring a number of alternative solutions for different needs, including new mobile apps that are capable of WordPress administration. With JSON support natively built into both iOS and Android, the monopoly of the [WordPress apps](#) is no longer present. I am really excited to see if someone can come up with something exceptionally innovative.

Third-Party Content Available Within WordPress

What's more, since the API makes WordPress content independent of programming language and environment, you will now be able to source material from anywhere and display it in any way you want.

It even gives Ruby on Rails and other frameworks access to WordPress functionality and vice versa. Advances in these

areas will therefore become available for WordPress users. This is especially good news for front-end developers, as it enables them to do whatever they want on the front part of their site without having to deal with the back-end.

For non-developers, on the other hand, this technology can be used to create new plugins and solutions to integrate content from other web profiles on their WordPress site.

Something that I am personally interested to see is if the API will open the door for sending content to WordPress directly from video editing, graphic design, and other third-party tools.

WordPress Moves into New Spheres

Not only do new types of content become available for WordPress, but with the new API, the inverse is just as true. Systems not built on PHP will be able to pull WordPress data to use it within their own applications.

However, what might be most interesting is the possibility of porting WordPress products (such as plugins) to other platforms and content management systems.

While WordPress already powers one quarter of the web, imagine being able to bring your products to the remaining 75 percent!

WordPress-Based Mobile Apps are Becoming a Thing

Finally, the WordPress REST API is also great news for anyone using or looking to build mobile apps. I am guessing since apps continue to be the preferred reason for smartphone usage, there should be a few of you out there.

With a RESTful API, WordPress could now become the glue that holds apps together as is demonstrated by this [AppPresser tutorial](#).

I expect to see solutions for everyday users that help them to automatically create mobile apps based on WordPress sites. The aforementioned JSON support of the two largest mobile operating systems should only make this easier.

FURTHER READING

The above is just the tip of the iceberg. The future will show what other ideas smart developers can come up with for the WP API.

If you've been following along so far and are looking for more info on the WordPress REST API, here are a few choice resources. Some of them are more technical and aimed at developers, but if you are interested in understanding the background, I'd encourage you to read them anyway:

- [Tutsplus - A Beginner's Guide to HTTP and REST](#)
- [CSS-Tricks - Using the WP-API to Fetch Posts](#)
- [WP REST API Documentation](#)
- [WP REST API Guides](#)
- [Josh Pollock - The Ultimate Guide To The WordPress REST API](#)

SUMMING UP

As you can see, the new WordPress REST API will transform WordPress as we know it. It will open up the platform to more influence from the outside and deepen collaboration with third-party applications and services.

The WordPress REST API is not just exciting news for developers. Non-technical users can look forward to new products and possibilities as well as stronger integration of different types of content. The API will allow for some completely new applications of the WordPress platform. I am excited to see what the community will make of it (and you should be too!).

ABOUT THE AUTHOR



Nick Schäferhoff

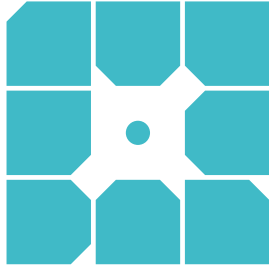
Nick Schäferhoff is an entrepreneur, online marketer, and professional blogger from Germany. He found WordPress when he needed a website for his first business and instantly fell in love. When not building websites, creating content or helping his clients improve their online business, he can most often 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](#).



About Torque

Torque is a news site featuring all things WordPress. We are dedicated to informing new and advanced WordPress professionals, users, and enthusiasts about the industry. Torque focuses primarily on WordPress News, Business, and Development, but also covers topics relating to open source and break-through technology. Torque made its debut in July 2013, at WordCamp San Francisco, and has since produced valuable content that reflects the evolution of WordPress, both as a platform and a community. Torque is a WP Engine publication, though maintains complete editorial independence.

torquemag.io



About WP Engine

WP Engine is a leading SaaS content management platform for websites and applications built on WordPress. The company powers thousands of websites and apps built on the WordPress platform delivering a fast, reliable and secure web experience. All levels of users including bloggers, marketers, SMBs and large corporations rely on WP Engine's platform to keep their websites up and running. The company's exceptional customer service team specializes in quickly solving technical problems, and creating a world-class customer experience ensuring that each user's WordPress site continues to perform at its full potential. Founded in 2010, WP Engine is headquartered in Austin, Texas and has offices in San Francisco, California, San Antonio, Texas and London, England.

wpengine.com

TORQUE

ALL THE WORD THAT'S FIT TO PRESS

A WP ENGINE PUBLICATION

torquemag.io

editor@torquemag.io