WP ROCKET
Superior WordPress Performance

☰

What is OPcache and How Do You Use It?

🔍   Page speed and caching ▾

Table of Contents

Published on October 1, 2019 - Updated on October 29, 2019

You can speed up your WordPress site so it's around three times faster or more with the OPcache PHP OPcode caching system.

OPcache is a type of caching system that saves precompiled script bytecode in a server's memory called a cache, so each time a user visits a web page, it loads faster.

Here's more detail on OPcache and how to install it for your WordPress site to speed it up.

What is Caching?

Caching is a system you can put in place to speed up your site. It works by saving content to your server's memory the first time it's loaded on a web page. Each subsequent page load has the stored content retrieved from memory and served on the page.

This process means cached content is displayed a lot faster than if it's loaded directly from the server.

It's like memorizing your multiplication tables. Once you have memorized it, it's so much faster to recite the answer to a multiplication problem from memory rather than trying to calculate the answer all over again.

A cache works in a similar way. Content is stored in a server's memory so it can be loaded from there quickly instead of going all the way to the server to load the content which takes more time.

The result being a faster loading WordPress website.

There are also **different types of caching such as browser, site, object, and OPcode caching**. It's recommended that you implement more than one kind to increase your site's performance.

For details, you can also check out these resources:

- **Caching for WordPress, Explained in Plain English**

- **Browser Cache vs Cookies: What's the Difference?**

- **What is Object Caching and How to Use It With WordPress**

What is OPcache or PHP Opcode Caching?

OPcache is a type of **OPcode caching.** This kind of caching compiles human-readable PHP code to code your server understands which is called opcode. This occurs when the PHP file loads on a web page for the first time. Then, it's saved to the server's memory for faster loading at each subsequent page visit.

Bytecode cache engines such as **OPcache**, **APC**, and **Xcache** all complete this process the first time the PHP file is executed without having to do it a second, or third time.

How PHP Opcode Caching Works

When a PHP script executes, your server's cache memory is checked to see if the script has already been cached. If it hasn't, it's parsed, which means the code is analyzed.

Then, the script is compiled into opcode making the file readable by the server. Once that's done, the opcode is saved to the server's memory.

In other words, it's stored in your server's cache. The next time a visitor loads the page with the PHP script, the cached code is executed and loaded much faster.

On the other hand, if the script is loaded on the page and the cache is checked for opcode and finds it, then it's loaded lickety-split.
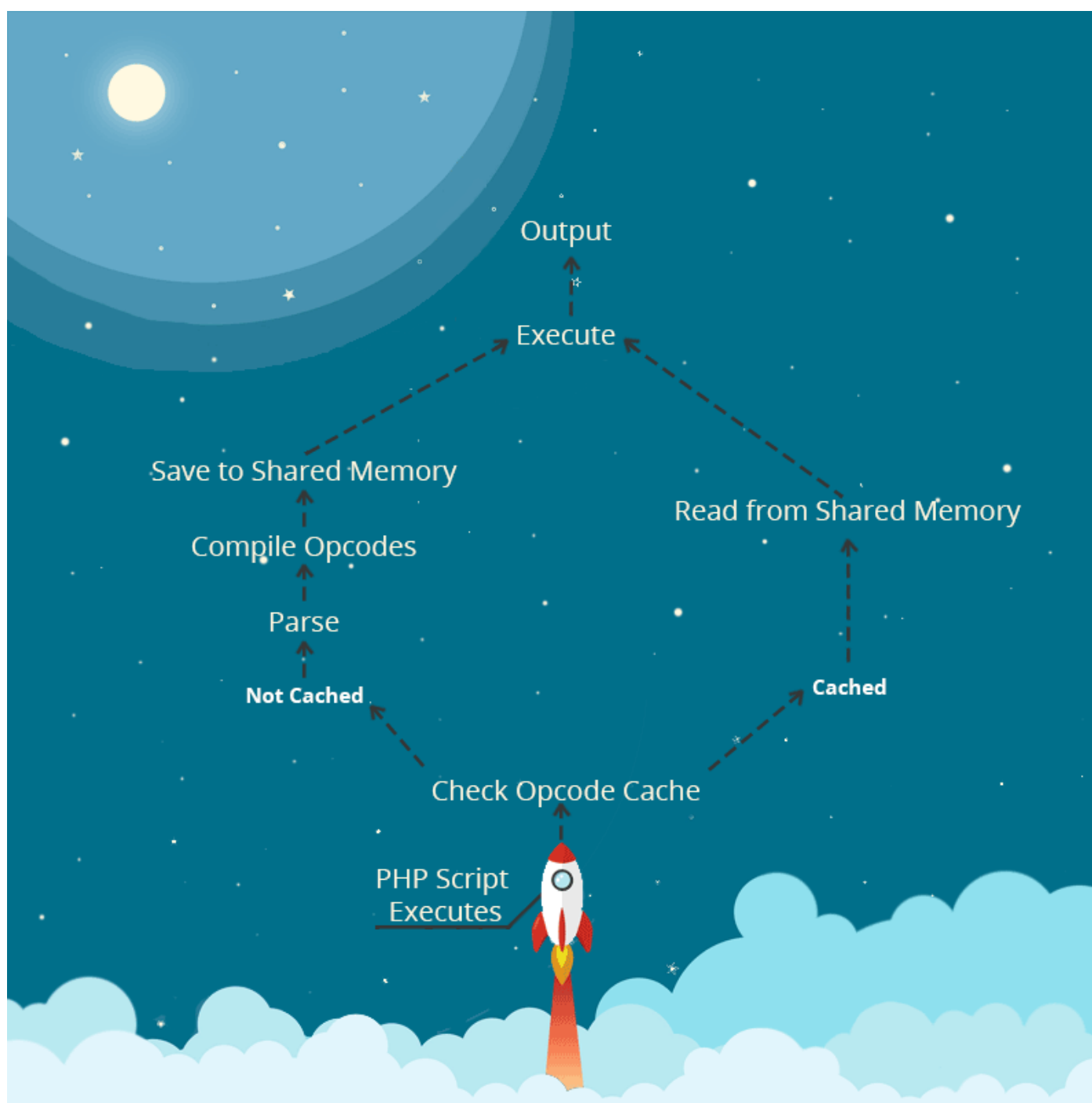
When PHP scripts aren't in the cache, they're cached for subsequent page loads.

The Differences Between OPcache and APC Caches

OPcache, APC as well as Xcache are all opcode caching systems. OPcache used to be owned by Zend and Alternative PHP Cache (APC) was a free, open source extension for PHP. Xcache was also an alternative option.

APC was widely used, but it didn't have the backing that OPcache had so it could be well maintained and stable with each new PHP release.

Fortunately, Zend made OPcache open source and available as an extension since PHP version 5.5. In earlier versions, you have the choice to use APC or OPcache, but if you would like to use the latter, you need to manually install it.

Xcache, on the other hand, is a good alternative to OPcache as a PHP accelerator.

Will OPcache Speed up My WordPress Site?

All three options are suitable for WordPress, but **the recommended option for PHP versions 5.5 and above is OPcache**. On average, it speeds up WordPress three fold, at the very least for medium to large sites.

If you have a smaller site without many additional PHP scripts or plugins installed, you likely won't notice much of a difference.

However, you can still install OPcache on even small WordPress sites with no negative effects other than a slight increase in memory usage. But, it won't be enough to cause any issues. This is similar for medium, large, or enterprise sites.

How to Install OPcache on Your Server

If you have **PHP version 5.5 and above**, OPcache PHP opcode caching is installed and enabled by default. You don't have to do anything else. There's also no additional requirements or configuration needed to run it.

That's also why you won't find any options if you were to look for them.

For details, check out **Why You Need to Upgrade to PHP 7+ ASAP. (and How to Do It Right Now)**

According to the PHP manual:

> 66
>
> "OPcache can only be compiled as a shared extension. If you have disabled the building of default extensions with **–disable-all,** you must compile PHP with the **–enable-opcache** option for OPcache to be available.
> Once compiled, you can use the **zend_extension** configuration directive to load the OPcache PHP opcode caching extension into PHP. This can be done with *zend_extension=/full/path/to/opcache.so* on non-Windows platforms, and *zend_extension=C:\path\to\php_opcache.dll* on Windows."

When you have done that, restart PHP using SSH.

On Apache, enter the command below to restart PHP.

```
1   apache2ctl restart
```
Restart PHP Apache hosted with ♥ by **GitHub**   view raw

For Nginx, enter the following:

```
1   service nginx restart
```
Restart PHP Nginx hosted with ♥ by **GitHub**   view raw

How to Install OPcache on Earlier Versions

If your server is running on PHP versions 5.2, 5.3, or 5.4 you can manually install OPcache using the **PECL** command below:

```
1   pecl install zendopcache-beta
```
Install OPcache PHP v5.2-5.4 hosted with ♥ by **GitHub**   view raw

Next, go to you *php.ini* file:

```
1    pecl config-get php_ini
```
Go to php.ini hosted with 🖤 by **GitHub**   view raw

You'll need to update your *php.ini* file with the following recommended settings:

```
1    opcache.memory_consumption=128
2    opcache.interned_strings_buffer=8
3    opcache.max_accelerated_files=4000
4    opcache.revalidate_freq=60
5    opcache.fast_shutdown=1
6    opcache.enable_cli=1
```
**php.ini** hosted with 🖤 by **GitHub**   view raw

You can often find your *php.ini* file among your site's file folder system. If you're not sure how to find it, contact your hosting provider.

Clearing Opcache with WP Rocket

Once you have OPcache or PHP opcode caching installed, you may need to clear the cache. You can do this in one click using **WP Rocket**.

Once the plugin is installed on your WordPress site, go to **Settings > WP Rocket** and click the **Purge OPcache** button. You can find it on the right side of the page.
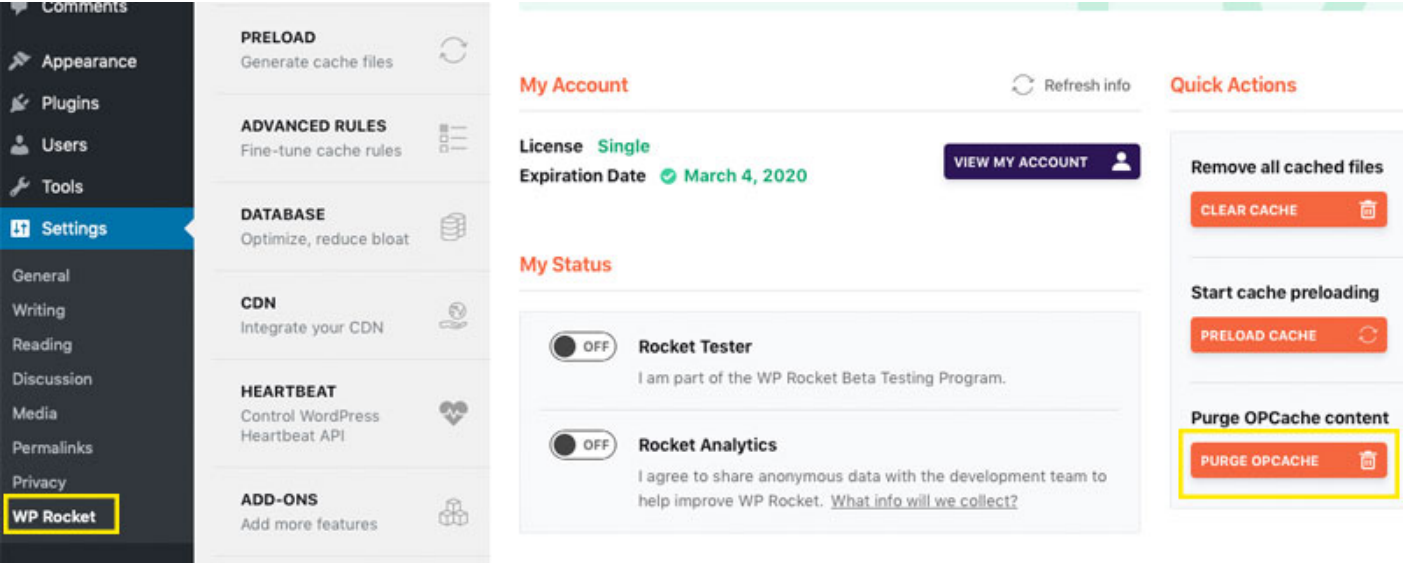
You can purge OPcache in one click with WP Rocket.

You can also check out **What is OPCache?**

Wrapping Up

For many WordPress site owners, their server may already have the latest version of PHP installed. This means they already have OPcache automatically enabled to drastically speed up page load times for their site.

For those who have PHP version 5.2 to 5.4, you can manually install OPcache with the steps outlined above.

Do you have the latest PHP version installed? Do you have any questions about OPcache/PHP opcode caching? Let us know in the comments below.