Supported PHP versions: 7.4 to 8.5

Which PHP versions Exponential 6 supports and how each is tested - PHP 7.4 as the oldest (8.0 for Velocity), 8.5 as the newest, the composer.json constraint, the CI jobs per version, the differences on 8.0, and what to...

Source file: `doc/eng-us/help/php-developers/supported-php-versions.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/php-developers/supported-php-versions.md

> **Supported: PHP 8.0 to 8.5. Usable without support: PHP 7.4 (Exponential 6.0.15 and later).** Why Exponential supports PHP 7.4

This page is for PHP developers who need to know which PHP versions their code must run on. Exponential 6 runs on PHP 8.0 to 8.5 (PHP 7.4 usable without support). The oldest version decides what syntax and functions you may use; the newest decides which deprecations you must avoid.

Contents

1. The range
2. PHP 8.0 supported, PHP 7.4 usable without support
3. How each version is tested
4. What differs on PHP 8.0
5. The history of the requirement
6. What to declare in your extension

The range

`composer.json` declares:

JSON

```
"php": "^7.4 || ^8.0 || ^8.1 || ^8.2 || ^8.3 || ^8.4 || ^8.5 || ^8.6 || ^8.7 || ^8.8"
```

The ranges for 8.6 to 8.8 look forward; those versions do not exist yet and are not tested (PHP, PHP-FIG and Composer standards). The required PHP extensions are `dom`, `libxml`, `mbstring`, `pcre`, `json`, `iconv`, `reflection`, `session`, `spl` and `simplexml`; database and image extensions are in `suggest`.

Exponential Velocity, the application server, runs on PHP 8.0 to 8.5 from its release 0.0.4.54 (Velocity is part of Exponential 6.0.15). The kernel requires it, together with the other packages that need PHP 8, through the metapackage `se7enxweb/exponential-php-floor`, whose version for PHP 7.4 leaves them out (Why Exponential supports PHP 7.4).

PHP 8.0 supported, PHP 7.4 usable without support

PHP 7.4 is supported from 6.0.15 for organizations that cannot change the PHP of their servers yet: hosting contracts, certified images, long-term operating system releases. PHP 8.0 matters because it is the stock PHP of Red Hat Enterprise Linux 9, AlmaLinux 9 and Rocky Linux 9, so a

site on those systems needs no third-party PHP packages. The reasons and the trade-offs are in Why Exponential supports PHP 7.4.

The cost for developers is a rule: no syntax and no unguarded functions newer than 7.4 in the kernel. That means no union types, nullsafe operator, `match`, named arguments, constructor promotion, `mixed`, `static` return types, enums or `readonly`; typed properties and arrow functions are fine. Functions of PHP 8.0 and 8.1 such as `str_contains()` may be used, because `symfony/polyfill-php80` and `symfony/polyfill-php81` define them on 7.4 (Writing code that runs on PHP 8.0 lists the rest). `make lint74` checks the files you changed, `make lint74-all` every file.

How each version is tested

Check	PHP	Where
Syntax of every PHP file, production `composer install --no-dev`, boot smoke test (`tests/bin/bootsmoke.php`)	7.4	`quality.yml`, job `php74`
Syntax of every PHP file, production `composer install --no-dev`, boot smoke test (`tests/bin/bootsmoke.php`)	8.0	`quality.yml`, job `php80`
PHPCompatibility with `testVersion 8.0-`	static	`phpcs.xml.dist` (a warning in CI)
PHP syntax of changed files	8.3	`quality.yml`
PHPUnit without a database	8.1, 8.2, 8.3, 8.4, 8.5	`phpunit.yml`, with PHPUnit 10.5 on 8.1, 11.5 on 8.2 and 8.3, 13 on 8.4 and 8.5
PHPUnit with MySQL 8.0 and PostgreSQL 15	8.3	`phpunit.yml`, database tests
PHPStan (analysis as `phpVersion: 80100`)	8.3	`quality.yml`
Micro benchmark	8.5	`performance.yml`

PHPUnit does not run on PHP 8.0, because PHPUnit 10 and later need 8.1. On 8.0 the code is covered by syntax, install and boot only, which is why the rules for 8.0 matter so much in review (CI and reproducing a red check).

What differs on PHP 8.0

The kernel behaves the same on every version; a few features need newer PHP and fall back quietly:

Feature	On PHP 8.0
`array_is_list()`	defined in `lib/phpcompat.php`
`fputcsv()` with `$eol`	written without `$eol`, line endings replaced afterwards
`fsync()`	behind `function_exists()`; the data is flushed with `fflush()`
`memory_reset_peak_usage()`	behind `function_exists()`
`str_increment()`, `str_decrement()` (8.3)	the string operators return their input
`hash( 'xxh128' )`	the SQL query cache uses md5
AVIF image output under Velocity	needs 8.1; negotiation stops at WebP

The full table with the files is in PHP 8.0 support.

The history of the requirement

Release	Requirement
up to 6.0.7	PHP 7.4 and 8.1
6.0.8 (April 2025)	Composer requires PHP 8.1 or newer; initial 8.4 support
6.0.11 (December 2025)	PHP 8.5 support
6.0.15	PHP 8.0 supported again, checked in CI; then PHP 7.4 usable again without support, with the metapackage `se7enxweb/exponential-php-floor` and a CI job `php74`

Every step is listed in PHP 8 support.

What to declare in your extension

- Declare `"php": "^7.4 || ^8.0"` (or a list like the kernel's) in your extension's `composer.json` unless your extension really needs more; `"php": "^8.0"` is the next choice. A higher minimum blocks installation on RHEL 9's stock PHP, and the kernel's CI jobs `php74` and `php80` fail when a package the kernel requires directly raises its minimum above theirs. A package that must need PHP 8 belongs in the matching versions of `se7enxweb/exponential-php-floor`, not in the kernel's `require`.
- If you need a newer PHP feature, guard it as the kernel does instead of raising the minimum.
- Test on the oldest and the newest version you declare.

Related pages

- Writing code that runs on PHP 8.0
- PHP 8 migration notes for your code
- Modernizing composer.json
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.