PHP 8 support

Source file: doc/eng-us/bc/6.0/php8.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/bc/6.0/php8.md

Read this page if you upgrade from a release before 6.0.0, or if you maintain your own extensions. The kernel and libraries were changed step by step for PHP 8.0 up to 8.5 between December 2023 and April 2026. Compatibility functions were kept wherever they were known to be needed, so custom extensions do not end in fatal errors. To get rid of deprecation warnings, you may still have to adapt your own code, most often classes that extend `eZPersistentObject` or `eZDataType`.

In short

What changed	PHP 8 syntax and checks throughout the kernel; since 6.0.8, Composer requires PHP 8.1 or newer. Since 6.0.15 the code runs on PHP 8.0 again (PHP 8.0 support), and on PHP 7.4 (Why Exponential supports PHP 7.4).
Who is affected	Sites still on PHP 7.4 (6.0.8 to 6.0.14 need PHP 8.1; 6.0.15 is usable on 7.4 again, without support); authors of extensions with dynamic properties, unchecked `count()` or `null` passed to string functions.
How to check	Enable the debug output on a development copy and look for "Deprecated" notices from your extension files.
How to fix	Follow "What this means for your own code" below.

The sequence of changes (December 2023 to April 2026)

The first column is the first release that carries the change. Use it to see which of your extensions may need the same treatment.

Release	PHP topic	What changed
6.0.0 (Dec 2023)	8.3	Fixes for `eZTemplateForFunction` (the `{for}` template function) and further PHP 8.3 incompatibilities; the setup wizard no longer prints PHP 8 warnings during installation; the installation autoload code was rewritten so no error text appears.
6.0.1 (Jan 2024)	8.0	`count()` is called only after `is_countable()` in places that received `null` or scalars; inconsistent testing of data before use fixed. Found while testing the SQLite driver.
6.0.4 (Aug 2024)	8.2	Dynamic properties of `eZPersistentObject` classes no longer cause warnings in the debug output.
6.0.11 (commit of Dec 2024, first in 6.0.11)	8.0	The `ezmatrix` datatype no longer ends in a fatal error (`f433a08cf9`; the commit sits on a branch that reached a release only with 6.0.11). A smaller PHP 8 notice fix of December 2024 (`21d2c6b4f5`) is in 6.0.6.
6.0.7 (Jan 2025)	8.0 and 8.2	The template `{for}` loop warning, the star rating extension and PostgreSQL library warnings fixed; unlinking an already removed file no longer warns.

Release	PHP topic	What changed
6.0.8 (Apr 2025)	8.4	The removed `E_STRICT` constant is no longer referenced; initial 8.4 support in ten kernel and library files (`eZSession`, `eZINI`, `eZExtension`, `eZSys`, `eZOrder`, `eZDebug`, `eZContentObject`, the autoload generator and others, `86abc11907`). **Composer now requires PHP 8.1 or newer** (`^8.1 || ^8.2`); PHP 7.4 is no longer installable.
6.0.11 (Sep 2025)	8.1 and 8.2	`#[ReturnTypeWillChange]` on methods that implement SPL interfaces (no behaviour change); `utf8_decode()` in `ezldapuser` replaced (deprecated in 8.2); public properties declared on `eZURL` and other classes instead of dynamic ones; `null` is no longer passed to `mysqli_real_escape_string()` and `preg_match()`.
6.0.11 (Dec 2025)	8.5	Nullable types where PHP 8.5 requires them; Zeta Components raised to releases that support PHP 8.5; the README states PHP 8.5 support.
6.0.13 (Feb 2026)	8.4	Time datatype and `eZOrder::fetchList()` deprecations fixed, see the hardening specification.
6.0.15 (Oct 2026)	8.0	PHP 8.0 is the oldest supported version (RHEL 9): `array_is_list()` defined when missing (`lib/phpcompat.php`), `fputcsv()` without `$eol` on 8.0, the HTML escaping functions get the PHP 8.1 default flags explicitly so `wash` escapes single quotes on 8.0 too, a PHP 8.0 syntax check and boot test in CI. See PHP 8.0 support.
6.0.15 (Oct 2026)	7.4	PHP 7.4 is usable again, without support, as a bridge: the kernel parses and runs on it, Composer picks the packages per PHP version through `se7enxweb/exponential-php-floor`, CI job `php74` (Why Exponential supports PHP 7.4)
6.0.13 (Apr 2026)	8.x	The session handler: `read()` returns an empty string for an unknown session, `gc()` uses `time()` and a start time, and the current siteaccess global is checked before use.

How to check your installation

Shell

```
php -v
grep -n '"php"' composer.json
```

The first line of `php -v` must show 8.0 or newer, also when the site runs on Exponential Velocity (from its release 0.0.4.54; see PHP 8.0 support). The `composer.json` line shows the PHP versions the current release accepts.

What this means for your own code

- **Declare the properties your classes use.** PHP 8.2 deprecates creating them on the fly.
- **Test before you count.** Use `is_countable()` or `is_array()` before `count()` or `foreach`.
- **Do not pass `null` to string functions.** Use `$value ?? ''`.
- **Mark SPL methods.** Give methods that implement `Iterator`, `ArrayAccess` or `Countable` the `#[ReturnTypeWillChange]` attribute, or real return types.
- **Raise your PHP constraint.** Set your own `composer.json` PHP constraint to `^8.0` or newer, the oldest version the product supports again since 6.0.15 (6.0.8 to 6.0.14 required `^8.1`). Exponential Velocity runs on 8.0 too from its release 0.0.4.54. Sites on PHP 7.4 can use 6.0.15 again, without support; 6.0.8 to 6.0.14 need PHP 8.1.

The PHP manual lists every incompatible change per version:

- PHP 8.0 incompatible changes
- PHP 8.1 incompatible changes
- PHP 8.2 incompatible changes
- PHP 8.3 incompatible changes

Related pages

- Security hardening
- PHPUnit 10, PHPUnit 13 and PHPUnit 13 on PHP 8.4
- `ezpSessionHandlerDB` PHP 8 compatibility bugfixes and tests
- PHP 8.4 and 8.5 for the older Symfony stack: the framework forks
- History: April 2025, September 2025, December 2025, February 2026, March 2026

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.