

PHP semantic versioning checker

Tom Rochette <tom.rochette@coreteks.org>

December 21, 2025 — [77e1b28a](#)

1 Problem

I use [semantic versioning 2.0.0](#) for my PHP libraries and I'd like to know when I generate breaking changes to my code. I would like to have a tool that tells me when that happens, as soon as possible so I can avoid creating backward-incompatible changes.

2 Solution

In 2015 I was working on a lot of PHP code and releasing various libraries as well as using a lot of libraries which were not always respecting the semantic versioning 2.0.0 rules. I understood that it was difficult to keep track of all the changes done to a codebase and that you needed a certain level of expertise to tell what kind of semantic versioning impact changes had. I knew that most of the semantic versioning rules however were rules that could be codified such that you could run a program that would look at a before and after snapshot of a piece of code and tell you how it changed. Those changes could then be categorized according to what they changed, and based on the semantics of semantic versioning, a semantic versioning change could be generated for the code change itself.

Thus [PHP Semantic Versioning Checker](#) was born. It analyzes a before and after snapshot of a directory of source code and generates a report of all the changes that occurred. It is then possible to review all the changes by their type (class, function, method, trait, interface) and their semantic impact (major, minor, patch, none). It also computes a suggested versioning change, which is the highest semantic impact found amongst the different types inspected.

I hope that this library and the concept of scanning source code for semantic versioning changes becomes more mainstream, such that every project runs this kind of tool as part of their CI pipeline.

3 References

- <https://github.com/tomzx/php-semver-checker>