

composer-tiki-org

<https://composer.tiki.org> [↗](#) supplies source code (external dependencies) that is bundled in Tiki, whereas [packages-tiki-org](#) is for code that is not bundled (An additional step must be done on each Tiki instance). This is known as [Tiki Packages](#).

They both replace getting the code from <https://packagist.org/> [↗](#) (which is the usual default for PHP development). Tiki has a [Long Term Support](#) cycle and this insures the code remains easily available to bundle in Tiki even if the original source is no longer available from Packagist.org (which is bound to happen when you have over 150 dependencies and a 5 year support period)

<https://composer.tiki.org> [↗](#) is

- hosted on a server managed by the Tiki community
- offering packages for multiple Tiki branches/versions, including some old ones
- available both in [http](#) [↗](#) and [https](#) [↗](#) because some users require http (yes, https is better)
- powered by [Satis](#) [↗](#)
- built several times per day from <https://gitlab.com/tikiwiki/tiki/-/blob/master/doc/devtools/satis.json> [↗](#)
- Packages are mainly copied from:
 - <https://packagist.org/> [↗](#) (preferred)
 - <https://asset-packagist.org/> [↗](#) (second choice)

To start using a new dependency, please make two merge requests:

1. One to <https://gitlab.com/tikiwiki/tiki/-/blob/master/doc/devtools/satis.json> [↗](#)
2. And then to `vendor_bundled/composer.json` and `vendor_bundled/composer.lock` for the relevant branch

Why two merge requests? Doesn't this go against the concept of the [Atomic Commit Convention](#)? Yes, but `satis.json` needs to first be modified so package can appear on `composer.tiki.org`, and then, a few minutes later, it becomes available to be used.

Tips:

- [How to add new versions without removing old versions](#) [↗](#). See also [here](#) [↗](#)

Related links

- [The Tiki 27 plus Build System](#)
- <https://getcomposer.org/doc/articles/versions.md> [↗](#)
- [External Libraries](#)
- [Composer](#)
- [How to pick a software library](#)