

Open Hub

Here are the stats: <https://www.openhub.net/p/tikiwiki>

Below are some explanations for major variations.

- In November 2023, the code base shrunk by 700 000 lines of code thanks to a code re-organization done along the introduction of a new build system. A lot of CSS was previously compiled and tracked in the source code (artificially inflating stats). Instead, it will be generated when needed.

In 2023-07, there was a question ("Why? More enthusiastic coders or more use of AI?") on Facebook and I am copying question and answers here for posterity:

It's a combination of things:

- Tiki25 was the biggest release ever, so that was a lot of commits:

<https://tiki.org/article497-Biggest-Tiki-Release-Ever-Tiki-25>

- Moving to PHP 8.1 in Tiki26 was also a large number of commits: <https://doc.tiki.org/PHP8>

- We started using <https://dev.tiki.org/Using-GlitchTip-as-part-of-the-Tiki-development-process> so this leads to a lot of small fixes: remove mostly harmless error messages but also fix some real underlying issues.

- We recruited a lot of junior developers and after an adaptation period, the output in quantity and quality is ramping up.

So this combination of factors is not going to stay forever. Once PHP8 and error logs are in good shape, the commits there will go down. And we'll shift the energy to more substantive commits (adding a new feature can be 30x more work than fixing an error in the logs). And the junior devs will continue to improve.

Quantity and Quality: Beyond having a lot of commits, the quality is improving (the average commit is significantly better than the average of the current code base):

- We handled thousands of merge requests, and you can read all the thoughtful discussions:

https://gitlab.com/tikiwiki/tiki/-/merge_requests?scope=all&state=all

- Thousands of tests are ran of each commit:

<https://gitlab.com/tikiwiki/tiki/-/blob/master/.gitlab-ci.yml>

And we'll revamp our build system for Tiki27, among other things: <https://dev.tiki.org/Tiki27>

- 2023-05 saw a surge of commits thanks to [Using GlitchTip as part of the Tiki development process](#) with hundreds of small commits to add support for PHP8 and remove (mostly harmless) error messages.
- In 2020-06, we see a steep reduction of monthly contributors. This is of course caused by Covid-19. In October 2021, the situation is back to normal.
- In 2019, the Tiki community switched over from SVN to Git, and this leads to more work/output per commit, because multiple commits are squashed into one. So the real activity level is much higher than it looks on the chart.
- 2018-09 and 2018-10 were busy with preparation of Tiki19, mainly the upgrade to Bootstrap 4.
- 2007-11 to 2008-02: The number of contributors and commits drops to nearly 0, which doesn't fit the trend. It also doesn't seem logical given that in 2008-08, Tiki 2.0 was released in which "80 people contributed 6894 commits". I (Marc) have a vague recollection that we had an issue with the source control system, perhaps the [migration from CVS to SVN](#).

Tiki vs WordPress vs Drupal

https://www.openhub.net/p/_compare?project_0=Drupal+%28core%29&project_1=WordPress&project_2=Tiki+Wiki+CMS+Groupware

Top committers of all time

<https://www.openhub.net/p/tikiwiki/contributors>

Top committers of the last 12 months

https://www.openhub.net/p/tikiwiki/contributors?sort=twelve_month_commits

Top committers of all time, that are active in last 12 months

https://www.openhub.net/p/tikiwiki/contributors?sort=commits&time_span=12+months

Top committers of all time, that are active in last 30 days

https://www.openhub.net/p/tikiwiki/contributors?sort=commits&time_span=30+days

Tiki dependencies

We have a specific entry:

<https://www.openhub.net/p/tiki-dependencies>

Analysis

- In the 2013-2015 period vs 2017 and after
 - The number of commits is very high
 - The fluctuations are intense from month to month, which is surprising given it's aggregate data for over 100 projects.

Potential explanation: Many projects used today by Tiki were born in 2011-2013 and there was a lot of initial activity and things stabilized because many projects became more established and because of the move to Git (more work/output per commit, because multiple commits are squashed into one). Also, some projects were started on GitHub.

Comparison:

https://www.openhub.net/p/_compare?project_0=Tiki+dependencies&project_1=Tiki+Wiki+CMS+Groupware

Notes

- For charts, there are 2 possible values for time_span: 30+days and 12+months.

alias

- [Black Duck Open Hub](#) | [OpenHub](#)
-